Tampere University

Tuomas Mäenpää

OFF-PUCK SCORING OPPORTUNITIES

Detection and visualization of ice-hockey scoring opportunities from position data

Master of Science Thesis
Faculty of Information Technology and Communication Sciences
Examiners: Prof. Joni Kämäräinen
February 2024

ABSTRACT

Tuomas Mäenpää: Off-puck scoring opportunities
Master of Science Thesis
Tampere University
Degree Programme in Information Technology, MSc (Tech)
February 2024

The purpose of this thesis was to investigate if off-puck player positions can be utilized to predict goal-scoring in ice hockey. In addition to predicting scoring, the aim was to predict where the goals would be scored from. Scoring goals is the most important aspect of any ball-sport. In ice hockey, players spend most of the match without the puck, yet they continuously impact the course of the match. During attacking play, the players without the puck support the controlling player with their movement and positioning. Currently, there are no methods to quantify the positioning of off-puck players during attacking play. The objective of this thesis was to develop a method to detect and locate off-puck scoring opportunities based on position data.

The data used in this study was collected with Wisehockey's sports tracking system. The dataset contained all shot events and all successful pass events from 250 professional ice hockey matches during the 2022/2023 Liiga season. In addition to the events, full position data for the puck and players was available from those matches.

The Off-Puck Scoring Opportunity (OPSO) model was built from three separate probability models. The models estimated the probability of pitch control, the probability of a successful pass, and the probability of scoring. The output of the OPSO model was a probability density map that represented the probability of scoring within 5 seconds from the next on-puck event from any location within the rink. The pitch control probability was modeled with a parametric approach based on the position data of the players. The pass probability was modeled by assimilating the displacement of the puck between consecutive events. Three machine-learning methods were used to estimate the probability of scoring: exponential distribution, logistic regression, and a multilayer perceptron. The models were tasked to predict goal-scoring based on event location. The classifiers reached AUC scores between 0.81 and 0.83. The objective was to exceed the prediction quality with OPSO.

Two validation methods were used to test OPSO. First, a convolutional neural network (CNN) was trained and tested on the output heatmaps produced by OPSO. The CNN reached the AUC score of 0.72. Second, the total scoring probabilities were integrated from the output probability density maps. The AUC score for the integrated probabilities from OPSO was 0.84. The results showed that OPSO improved the ability to detect off-puck scoring opportunities. Taking players' positioning and movement into account in scoring probability estimation enhanced the prediction reliability. The objective of the thesis was met and the purpose was fulfilled.

Keywords: ice hockey, machine learning, scoring probability, pitch control, pass probability

TIIVISTELMÄ

Tuomas Mäenpää: Kiekottomien pelaajien tuottamat maalipaikat
Diplomityö
Tampereen yliopisto
Tietotekniikan DI-tutkinto-ohjelma
Helmikuu 2024

Tämän diplomityön tarkoituksena oli tutkia maalinteon ennustamista jääkiekossa kiekottomien pelaajien sijainnin perusteella. Lisäksi tavoitteena oli ennustaa mistä kaukalon sijainnista maali syntyy. Maalinteko on kaikkien palloilulajien tärkein osa-alue. Jääkiekossa pelaajat viettävät suurimman osan ottelusta ilman kiekkoa vaikuttaen kuitenkin jatkuvasti pelin kulkuun. Hyökkäysten aikana kiekottomat pelaajat tukevat kiekollista pelaajaa liikkumisellaan tarjoten syöttövaihtoehtoja. Tällä hetkellä ei tunneta menetelmiä pelaajien sijoittumisen laadun mittaamiseen hyökkäysten aikana. Tämän diplomityön tavoitteena oli toteuttaa menetelmä, joka tunnistaa ja paikantaa kiekottomien pelaajien tuottamia maalipaikkoja sijaintidatasta.

Tässä tutkimuksessa käytetty paikannusdata kerättiin Wisehockey urheiluanalytiikka-alustalla. Data sisälsi kaikki laukaukset sekä onnistuneet syötöt 250 Liiga-ottelusta kaudelta 2022/2023. Syöttöjen ja laukausten lisäksi saatavilla oli pelaajien ja kiekon paikannusdata näistä otteluista.

Tässä työssä toteutettu ”Off-Puck Scoring Opportunity” (OPSO) -malli koostettiin kolmesta erillisestä todennäköisyysmallista. Näillä mallinnettiin alueellisen kontrollin-, onnistuneen syötönsekä maalinteon todennäköisyyttä. OPSO-malli tuottaa tiheysfunktion maalinteon todennäköisyydelle mistä tahansa kaukalon pisteestä seuraavan viiden sekunnin aikana. Alueellisen kontrollin todennäköisyyttä mallinnettiin parametrisella menetelmällä perustuen pelaajien paikannusdataan. Syötön todennäköisyyttä mallinnettiin kiekon sijainnin muutoksilla peräkkäisten kiekollisten tapahtumien välillä. Maalinteon todennäköisyyden mallintamiseen hyödynnettiin kolmea koneoppimismenetelmää: eksponentiaalista jakaumaa, logistista regressiota sekä monikerroksista perseptroniverkkoa. Koneoppimismallien tehtävänä oli ennustaa maalintekoa kiekollisen tapahtuman sijainnin perusteella. Mallit oppivat erottamaan maaleihin johtavat tilanteet 0,81 - 0,83 ”area under the curve” (AUC) -metriikalla mitattuna. OPSO-mallin tavoite oli ylittää näiden mallien maalintekotilanteiden erottelukyky.

Mallin todentamiseen käytettiin kahta menetelmää. Ensin konvoluutioneuroverkko koulutettiin ja testattiin OPSO-mallin tuottamilla todennäköisyyslämpökartoilla. Neuroverkko pystyi erottamaan maaliin johtavat tilanteet 0,72 AUC-metriikalla mitattuna. Toisessa todennusmenetelmässä käytettiin OPSO-mallin tuottamista tiheysfunktioista integroituja maalinteon kokonaistodennäköisyyksiä. Näiden todennäköisyyksien erottelukyky oli 0,84 AUC-metriikalla mitattuna. Tulos osoittaa OPSO-mallin tunnistavan kiekottomien pelaajien maalipaikkoja tavanomaisia koneoppimismenetelmiä paremmin. Kiekottomien pelaajien sijainnin ja liikesuunnan huomioiminen parantaa maaliin johtavien tilanteiden tunnistamista. Tämän perusteella OPSO todetaan luotettavaksi menetelmäksi kiekottomien pelaajien tuottamien maalipaikkojen tunnistamiseen. Diplomityön tavoite todetaan saavutetuksi.

Avainsanat: jääkiekko, koneoppiminen, maalinteon todennäköisyys, aluekontrolli, syötön todennäköisyys

CONTENTS

1. INTRODUCTION

Creating scoring opportunities in ice hockey requires skill and intelligence from all participating attackers. The player with the puck must decide whether to carry the puck, pass it to a teammate, or take a shot. The attackers without the puck support the player with the puck through their movement and positioning. Evaluating the decisions made by the player with the puck is simple. Did the player lose the puck? Was their attempted pass successful? Did the shot they took lead to a goal? However, the actions committed by the off-puck players often go unrecognized. The scoring opportunities produced by off-puck players are only noticed if the player carrying the puck plays a pass to them. Furthermore, players are only in control of the puck for a fraction of their playing time. Currently, there are no methods for quantifying off-puck player positioning in terms of creating scoring chances. The aim of this thesis is to build an Off-Puck Scoring Opportunity (OPSO) -model to detect and visualize goal-scoring opportunities based on the positioning of the off-puck players at the instantaneous state of the game.

Professional sports are constantly becoming increasingly data-driven [1]. The performance of athletes is measured both in practice and during competition to gather insights into their physical development as well as their ability to perform in competition. In ice hockey, similarly to other ball sports, most traditional measurements of in-game performance are based on events such as passes and shots, which can be tracked manually. Current advancements in sports tracking technologies have opened up new possibilities for more sophisticated measures of continuous performance. There have been a number of methods proposed for spatial analysis of player performance presented for football. Papers published by Spearman and Fernandez in 2018 explore the possibility of measuring player performance given the spaces they occupy during matches [2, 3]. These papers serve as the foundation for this thesis. The research question is whether the positioning of off-puck players can be used to predict goal-scoring in ice hockey. Figure 1.1 shows an example of an off-puck player presenting a goal-scoring opportunity. The objective is to detect movements like this irrespective of whether the player controlling the puck executes the pass to the player in the best scoring position.

Chapter 2 reviews the evolution and state of sports data analytics today. Localization technologies, sports data tracking system Wisehockey, and machine learning algorithms

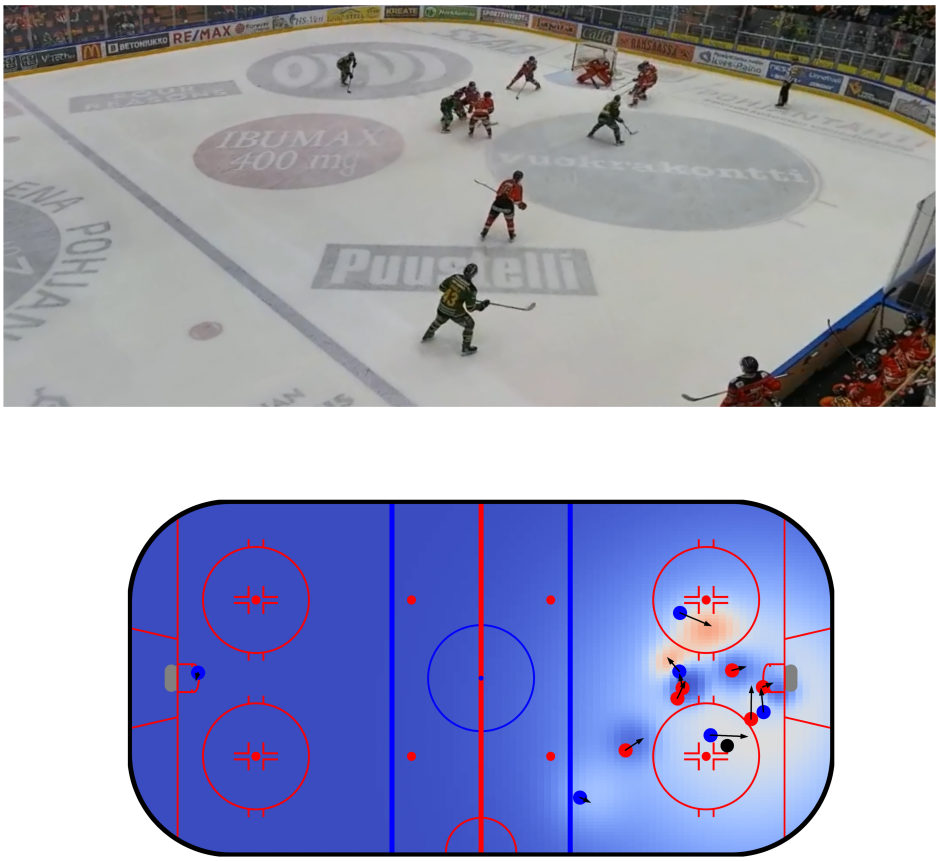

Figure 1.1. *An example of an off-puck player on the left faceoff circle presenting a goal-scoring opportunity. The lower graph shows how the OPSO model detects off-puck scoring opportunities.*

relevant to this thesis are explored in Chapter 3. The dataset used to conduct this study and its features are presented in Chapter 4. The OPSO model components are built in Chapter 5. Experiments conducted with the OPSO model are reviewed in Chapter 6 with possible applications for the model. The results of the thesis are summarized in Chapter 7 followed by future considerations for OPSO.

2. RELATED WORK

Professional sports organizations strive to find any advantages that will give them a decisive edge over competing organizations. In today's professional sporting industry, organizations have increasingly relied on data-driven insights to evaluate player performance, identify strengths and weaknesses, and make strategic decisions that can help them achieve success [1]. Michael Lewis created the "Moneyball" philosophy in his 2004 book with the same title [4]. The book explores how the Major League Baseball team Oakland Athletics used their limited resources efficiently by recruiting players who could produce performances higher than their salaries would have suggested. The Oakland Athletics general manager Billy Beane applied a data-driven approach to identifying undervalued players. The philosophy has since spread across the professional sports landscape since being able to find undervalued athletes in inefficient markets is attractive for any sporting organization. One motivation for this thesis is to present a method to detect goal-scoring opportunities that could serve as a base for player quality assessment methods.

Scoring is the most important aspect of ball sports. Depending on the sport, it is also a difficult metric for data analysis, especially if analysis needs to be conducted on a small number of matches given how infrequently scoring happens during matches. As a result, many analysts have proposed alternative methods to measure the quality of offensive and defensive performances during the match. In ice hockey, some of the most commonly used metrics are Corsi (all shots for minus all shots against) and Fenwick (unblocked shots for minus unblocked shots against) [5]. The advantage of Corsi and Fenwick compared to goals in an analytical sense is that they are based on shots, which are a much more frequent occurrence in ice hockey matches compared to goals. Thus, they can be considered a more reliable, less noisy, measurement of performance. However, not all shots are equally dangerous in terms of scoring. This leaves shot counts lacking important information about a team's offensive chance creation. Additionally, they fail to adequately reflect a team's ability to deny scoring opportunities for the opposing team.

During the past decade, the concept of expected goals (xG) has become a part of mainstream sports dialogue. Expected goals measures the probability of a shot resulting in a goal, given the features of the shooting opportunity. The concept of xG as a measurement of quality for shooting chances was introduced in ice hockey by Alan Ryder in 2004 in his

paper "Shot Quality, a methodology for the study of the quality of a hockey team's shots allowed" [6]. Ryder used his model to analyze differences between shooting chances and defensive capabilities between NHL teams during the 2002-2003 season. Ryder found that the isolated shot quality varies significantly between teams without correlation to shots on goal, meaning shot quality is an important metric for measuring teams' offensive and defensive output.

In 2012, Brian Macdonald presented his expected goals model for ice hockey in his paper "An Expected Goals Model for Evaluating NHL Teams and Players" [5]. Macdonald aimed to bring a solution for the difficulty of performance analysis due to the low number of goals in a single game. While he named the output of his model "expected goals", the model doesn't measure xG as it's currently understood. Macdonald's expected goals -model approximated the number of goals a team is expected to score based on several measured events in a game, including goals, shots, Corsi, Fenwick, zone starts, turnovers, and faceoffs. The model built from these features correlated with the number of actual goals scored better than any other individual metric.

The majority of advanced analytical models are built on event-based data. While many of those models have been accepted as a part of the decision-making process of the biggest sports organizations in the world, they leave an abundance of data completely untouched. Event-based data relies on the athletes deciding to commit an action. However, deciding not to take action is a decision that the athletes must make as well. Furthermore, the players not in control of the ball are also constantly making decisions on how and where they should move next. Many athletes have proven that having an elite physique can only take you so far if your ability to read the game is not at a high enough level.

With the rise of tracking data, there have been several proposals on how to measure the quality of players' positioning. In their 2018 article "Beyond expected goals", Spearman proposed a model they titled "Off-ball scoring opportunities" (OBSO) [2]. According to Spearman, OBSO can be used to "Identify and analyze important opportunities during a match, to assist opposition analysis by highlighting the regions of the pitch where specific players or teams are more likely to create off-ball scoring opportunities and to automate talent identification by finding the players across an entire league who are most proficient at creating off-ball scoring opportunities."

Spearman breaks down the match state into three components to evaluate the probability of a team scoring from any given location on the pitch. The first component is the likelihood of the team having control of the location. The second component is the likelihood of the team successfully moving the ball to said location. The third component is the likelihood of the team scoring from said location with the next on-ball event. Pitch

control is measured by modeling the time it would take for the players to reach location *p* on the pitch given their current location, speed, and movement direction. Pass success probability is calculated by the time it would take for the ball to travel to location *p*, how long it would approximately take for a player to control the ball, and whether opposing players have time to intercept the pass. Finally, the scoring probability is defined by the distance of location *p* from the goal. [2]

Fernandez et al. [3] presented their methodology to measure pitch control and the value of space in football in their paper "Wide open spaces". Whereas Spearman's approach aimed to measure the value of positioning directly related to scoring goals, Fernandez approached the question of space value from the perspective of build-up play. Fernandez proposed a novel, parametric approach to model pitch control where players' movement speed, direction, and distance from the ball define the influence they have on the area surrounding them. Controlling space has different values depending on the state of the game and how far up the pitch the space is controlled. Fernandez proposed a space value model where the most valuable spaces on the pitch are thought to be the spaces that the defending team is trying to protect the most. This was achieved through a feed-forward neural network which was used to predict the sum of influence given the location of the ball.

Another proposal for a pitch control model was presented by Wu and Swartz in 2023 [7]. Their approach was separated from Spearman's and Fernandez's approaches in that there were regions on the pitch that were not controlled by either team. In addition, their approach used an assumption of intent for players' movement. They assumed players move towards their intended targets as quickly as possible, which affected their probabilistic pitch control on their movement paths. Wu and Swartz faced the same issue as other pitch control methods in validating the model. They validated their pitch control model by comparing the pass succession rates to their proposed pitch control areas and found that their model predicted pitch control correctly for the receiver location in 91% of successful passes.

Spatiotemporal tracking creates new possibilities for performance analysis in sports. Players spend most of the matches without interacting with the ball, yet they are able to create scoring opportunities with their movement and positioning. Analyzing elite players' movements can bring new information about what players should strive for when they aren't in control of the ball. It can also bring additional information about players' quality to help organizations in their decision-making process. Lastly, new insight into the game can be used to bring fans a more in-depth view of the game and to spark new conversations about the abilities of players and teams.

3. THEORETICAL BACKGROUND

This chapter reviews the theoretical background for this thesis. First, the theory related to tracking technology and the tracking system used for this thesis is presented. Then, the machine learning methodologies used in this thesis are discussed. This includes an explanation of the metrics used to evaluate model performance at different stages of the thesis.

3.1 Indoor localization

Indoor localization refers to the procedure of finding the position of a user or a device in an indoor setting [8]. Indoor localization systems are used in various applications ranging from elderly persons care [9, p. 45] and industrial safety management [10] to indoor navigation in large buildings such as hospitals or universities [11].

Guyla reviews indoor localization techniques in their book "Recent Advances in Indoor Localization Systems and Technologies" [9]. Whereas outdoor positioning has reached a global standard in GPS, indoor localization systems have many competing methodologies that advanced greatly during the past decade. This has led to competing solutions for sensing, positioning, and tracking. The current indoor localization technologies can be divided into two core approaches: reference-based methods and reference-free methods. Reference-free localization methods use the traced object's inertial properties to estimate its position. Reference-based localization techniques work based on the target object sending a signal which is captured by some locators that use the signal to locate the object of interest. Reference-based localization methods rely on varying signals to create the reference system. Solutions range from audio signals and magnetic signals to visible light and radio signals, such as Wifi and Bluetooth.

There are numerous techniques to turn the reference signal into positioning. To calculate the position from the input signal, the signal needs to be detected and its features measured. Proposed measured properties of reference signals include the Received Signal Strength, the Time Of Flight, the Time Difference on Arrival, the Angle of Arrival (AoA), and the Angle difference on Arrival. [9, p. 1] Positioning based on the AoA depends on an array of locators receiving a reference signal and using the time difference of locators

receiving the signal to calculate the angle at which the signal arrives [8].

3.2 Bluetooth Low Energy

Gupta presents Bluetooth Low Energy (LE) as a technology designed to minimize the power usage of devices in their 2013 book "Inside Bluetooth Low Energy" [12, p. 135]. Bluetooth LE was introduced into the Bluetooth 4.0 specification. Devices using Bluetooth LE aim to be functioning for extended periods without the need to replace or recharge batteries. In addition, Bluetooth LE provides a small size, low cost, short range, fast connection option for radio transmission. Having small and cost-efficient tracking chips is essential for tracking sports data as the players don't want any extra weight on them not to mention that the ball or the puck shouldn't feel any different with the tracking chip in comparison to their regular counterparts.

3.3 Wisehockey system

Wisehockey is a real-time sports analytics platform that provides automatic statistics by tracking player and puck positions through matches. While the system can be used to track multiple different sports, this chapter will view the system from ice hockey's point of view. The system uses tracking chips which are placed within the puck and on the players' equipment to track their positions throughout matches. Several locators are placed above the rink to collect the Bluetooth LE signal sent by the tracking devices. The tracking data is collected on-site before being forwarded to the statistics calculation pipeline. This pipeline detects all events occurring in the matches and measures various metrics regarding the physical performance of the players. These metrics include the number of accelerations and decelerations made by a player during a game. A simplified version of the Wisehockey system architecture for ice hockey's use case is depicted in Figure 3.1. In addition to positional data, video recording is also captured from the matches to enrich the insights drawn from the tracking data.

The tracking system used is the Quuppa Intelligent Locating System. A set of Quuppa LD-7L locators are installed above the hockey rink. Each player wears a Quuppa QT1 tag on their shoulder pads during the matches. Practice sessions can be tracked with the system as well. The same tags are also embedded within pucks. These tags emit Bluetooth LE signals which are received by the locators. The local server in the arena runs Quuppa Management Engine which collects the AoA measurements and computes the location for each tag based on the AoA. The pucks send their signals at 50Hz frequency and the player-worn tags have a refresh rate of 20Hz as they can't reach velocities as high as the puck. The system's positioning accuracy is within a margin of less than 10 cm.

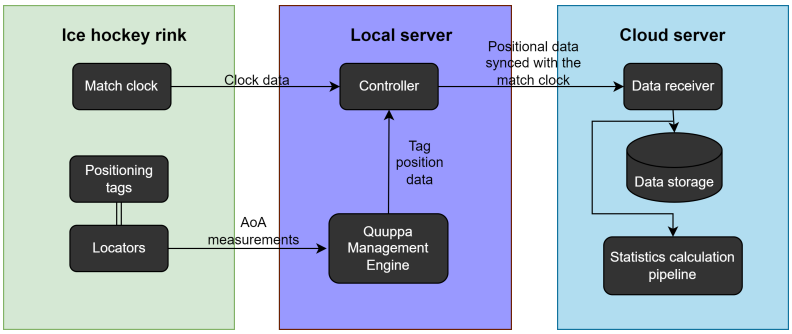

Figure 3.1. *The Wisehockey data gathering system architecture.*

The official match clock feed is integrated into the system to capture information about when the match is in progress. This way the system can accurately measure the events and movements only when the players are playing and ignore things that happen during stoppages. The match clock data is interlaced with the tracking data on the rink's local server. Unix timestamps are also added to the positional data and the clock data. The timestamped data is then streamed to the server running in the cloud.

On the cloud server, the data is received by the data receiver software. The position data is filtered before it is passed forward to reduce noise. Once the data is filtered and the snapshots have been ordered by their timestamps, the snapshots are inputted to the statistics calculation pipeline. The pipeline calculates the match statistics which are fed forward for archiving.

3.4 Machine learning

Machine learning is a field of computer science where predictive algorithms improve their performance iteratively by learning to generalize the training data. Machine learning algorithms are specifically useful for tasks where handling extremely large and complicated datasets would be highly impractical and cost-prohibitive. Machine learning algorithms are generally used for prediction and classification tasks as well as trend detection and finding underlying patterns in high-dimensional datasets. The following methods were used in the development of the OPSO model.

3.4.1 Exponential distribution

Exponential distribution is a probability distribution where the probability density decreases or increases exponentially as input parameter x grows. Exponential distribution belongs to the group of gamma distributions. Gamma distributions are defined by two parameters through one of two corresponding parametrizations:

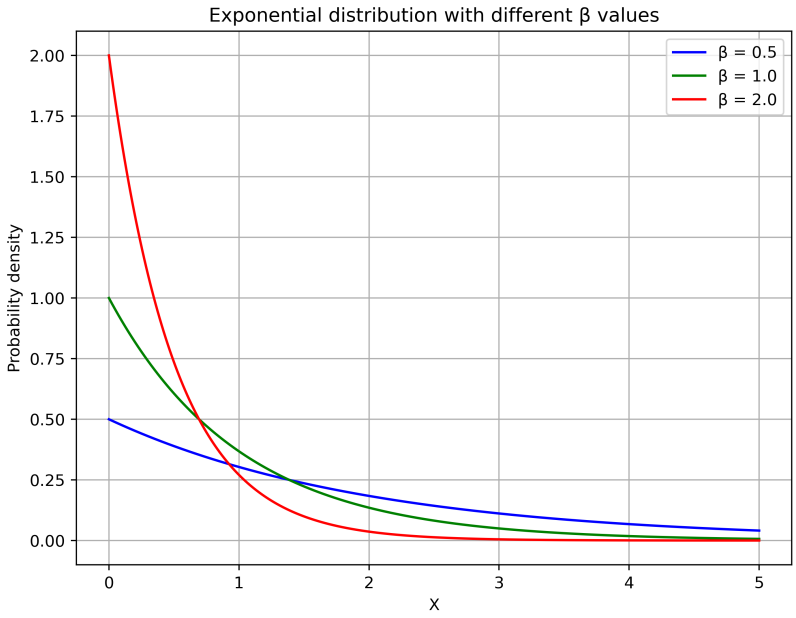

Figure 3.2. *Example of the probability density function of exponential distribution with different β values.*

1. Through a shape parameter *k* and a scale parameter α
2. Through a shape parameter $\alpha = k$ and a rate parameter $\beta = \frac{1}{\alpha}$

The probability density function of the exponential distribution is defined as

$$f(x;\beta) = \begin{cases} \frac{1}{\beta} e^{\frac{-x}{\beta}}, & x \geq 0 \\ 0, & x < 0 \end{cases} \tag{3.1}$$

where $\beta > 0$ is the rate parameter defining the steepness of the distribution. The exponential distribution is a special case of the gamma distributions as its shape parameter $\alpha = 1$. [13]

3.4.2 Logistic regression

Logistic regression is a linear classification model that is used for binary classification tasks. The main benefit of logistic regression models is that the model outputs probabilities for each predicted sample within [0, 1]. While logistic regression is inherently a binary classifier, it can be extended to multiclass issues by training multiple separate models to predict one-vs-rest for each class.

Logistic regression models the posterior probabilities of the *K* classes via linear functions in *x*. These probabilities have values within the range [0, 1] summing up to one. Logistic regression is defined

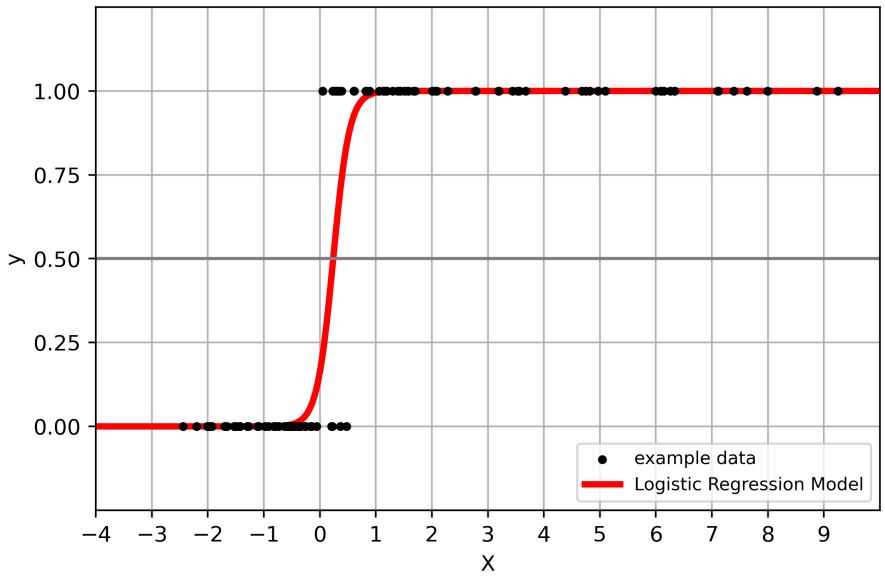

Figure 3.3. *An example of a logistic regression model fitted to data points. Figure adapted from source. [14]*

$$\begin{aligned}
\log \frac{Pr(G=1|X=x)}{Pr(G=K|X=x)} &= \beta_1 0 + \beta_1^T x \\
\log \frac{Pr(G=2|X=x)}{Pr(G=K|X=x)} &= \beta_2 0 + \beta_1^T x \\
&\vdots \\
\log \frac{Pr(G=K-1|X=x)}{Pr(G=K|X=x)} &= \beta_{(K-1)0} + \beta_{K-1}^T x
\end{aligned} \tag{3.2}$$

Logistic regression models are commonly fitted to data by maximum likelihood estimation, using the conditional likelihood of *G* given *X*. Given that $P(G|X)$ fully specifies the conditional distribution, the appropriate distribution to use is the multinomial distribution. The log-likelihood for *N* observations is

$$\lambda(\theta) = \sum_{i=1}^{N} \log p_{g_i}(x_i;\theta) \tag{3.3}$$

where $p_k(x_i;\theta) = P(G=k|X=x_i;\theta)$. [15]

3.4.3 Multilayer perceptron

Multilayer perceptron (MLP) is the standard architecture of an artificial neural network (ANN). MLPs consist of an input layer, a number of hidden layers, and an output layer. The layers in MLPs consist of individual perceptrons which imitate the functionality of

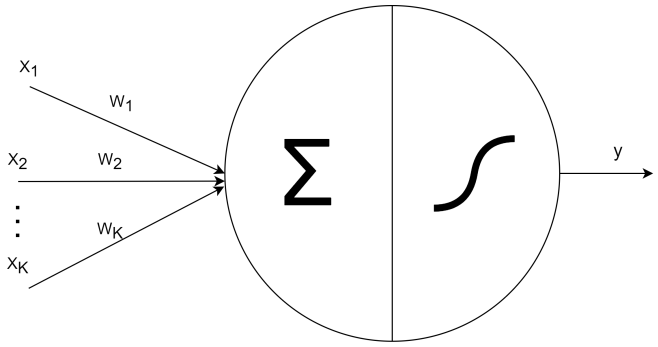

Figure 3.4. *The architecture of a single perceptron. Adapted from source. [16, p. 7]*

biological neurons. The architecture of a single perceptron is shown in Figure 3.4.

The perceptron operates by transforming input features through weighted summation and activation. An input array of K features is multiplied by a weight W before the multiplications are input to the summation function. A bias term is often included in the sum. The result of the sum is then used as an input to the non-linear activation function. The purpose of the non-linearity is to make the perceptron and a network of perceptron able to represent non-linear relationships between inputs and outputs [16, p. 7-8]. There are several popular activation functions used in artificial neurons. The most commonly used activation functions are Sigmoid function, Hyberbolic Tangent Function (Tanh), Rectified Linear Unit (ReLU), Exponential Linear Unit (ELU), and Softmax. The correct activation function to choose is dependent on the task at hand [16, p. 61]. The output of the activation function is also the output of the perceptron. A single-layer perceptron can function as a machine-learning algorithm on its own. In multilayer perceptron the network is built of at least three layers of neurons: an input layer, a number of hidden layers, and an output layer [17, p. 42].

The purpose of the input layer is to feed the input signal into the neural network. There are no calculations conducted in the nodes of the input layer [17, p. 86]. Once the signal has been passed through the input layer, each of its features is multiplied by a weight. The weights are unique for each feature and they vary across each node in the network. Figure 3.5 depicts an example MLP with two hidden layers. MLPs are fully connected meaning every node in a layer is connected to every node in the consecutive layer. Nodes in the hidden layers and the output layer apply calculations to the inputs they receive. The network learns by adjusting the weights and bias terms in the perceptrons through backpropagation.

In backpropagation, the network transfers the output it produces back to the hidden layers with the information about the correct output that the network should produce. The optimization is typically conducted by stochastic gradient descent where the vector of weights θ is moved in the direction of the negative gradient of a loss function L according to

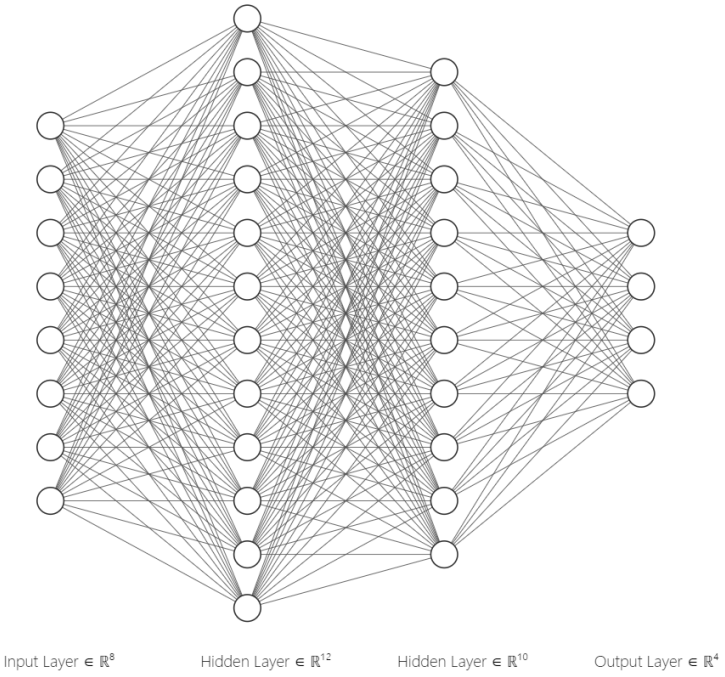

Figure 3.5. *An example of a multilayer perceptron with an input layer for a signal with 8 features, two hidden layers with 12 and 10 nodes, and an output layer with 4 nodes representing the 4 output values.*

$$\theta = \theta - \frac{\eta L}{\delta \theta} \tag{3.4}$$

where η is the learning rate. The backpropagation algorithm uses the chain rule to calculate the partial derivatives $\frac{\delta L}{\delta \theta}$. First, the input signal is propagated forward through the network to compute the pre-activations $z^{(l)}$ and activations $a^{(l)} = f^{(l)}(z^{(l)})$ for all the layers up to the output layer l_{n_l} where *f* is the transfer function of units. Second, the error at the output layer is calculated as

$$\delta^{(n_l)} = \frac{\partial L(a^{(n_l)}, y)}{\partial z^{(n_l)}} = \frac{\partial L(a^{(n_l)}, y)}{\partial a^{(n_l)}} \cdot f'(z^{(n_l)}) \tag{3.5}$$

where *y* represents the ground truth vector. In the next phase of the backpropagation algorithm the error is backpropagated to the lower layers $l = n_l - 1, n_l - 2, ..., 2$ by

$$\delta^{(l)} = ((W^{(l)\intercal} \delta^{(l+1)})) \cdot f'(z^{(l)}) \tag{3.6}$$

where $W^{(l)}$ represents the weight matrix of the layer *l* and layers are ordered so that $l = 1$ is the input layer and $l = 2$ is the first hidden layer. Once the error has been passed back

to each layer, the partial derivatives are calculated for the update:

$$\Delta_{W^{(l)}} L = \delta^{(l+1)} (a^{(l)})^\intercal$$
$$\Delta_{b^{(l)}} L = \delta^{(l+1)} \tag{3.7}$$

where $b^{(l)}$ is the bias vector of the layer *l*. Finally, the parameters are updated:

$$W^{(l)} = W^{(l)} - \eta \Delta_{W^{(l)}} L$$
$$b^{(l)} = b^{(l)} - \eta \Delta_{b^{(l)}} L \tag{3.8}$$

[18]

3.4.4 Convolutional neural networks

Convolutional neural networks (CNN) are a type of deep neural network that consists of preprocessing layers, where convolution and pooling operations are applied to the input data, and a fully connected layer. CNNs excel in image classification and object detection tasks but there are a number of applications for CNNs in other domains as well. In comparison to a standard fully connected network, convolutional neural networks can intake multidimensional data thus they're able to preserve structural information from the input data. This advantage of CNNs comes as a result of the adjusting of the convolutional masks in the convolutional layers through backpropagation. This adaptive filtering highlights different features and areas of interest from the input data. This chapter views CNNs from the perspective of 2-dimensional matrices, such as images.

Similarly to MLPs, the first layer in a CNN is the input layer, where the network receives the input signal without applying any calculations. Sewak et al. explain the convolution process in their 2018 book "Practical convolutional neural networks" [16, p. 31-33]. From the input nodes, the data is inputted into the first convolutional layer. In the convolutional layers, a square-shaped kernel of size $n * n$ is moved across the input matrix by a step of *x* elements. On each step, convolution is applied to the kernel and the values it covers. Applying convolution is the process of performing a dot product operation for the two matrices. This process is repeated until the kernel has passed through the entire input matrix. Depending on the implementation, different types of padding operations can be applied to the edges of the input matrix ensuring that the values on the edges can also be taken into account in the convolutions. The values on the sliding kernel are determined through gradient descent similar to the fully connected network backpropagation. Typically there are multiple kernels in each convolutional layer to capture different features, the number of kernels is usually above 10. Michelucci reviews how the data is shaped in the convolutional layers in their 2019 book "Advanced Applied Deep Learning Convo-

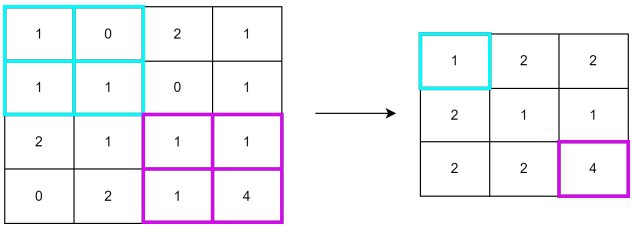

Figure 3.6. *An example of a Max-Pooling operation with window size (2, 2) and a step size of one element.*

lutional Neural Networks and Object Detection" [19]. The number of kernels determines the shape of the convolutional layer's output tensor. Bias term is added to the and an activation function is used to introduce non-linearity to the convolutional layers as well.

In order to reduce the dimensions of the data passing through the network, pooling operations are applied to it in pooling layers. Commonly, Max-Pooling is chosen as the pooling method. An example of how Max-Pooling produces its output is presented in Figure 3.6. In the example, a pooling kernel of size (2,2) is moved across the input matrix with a step size of 1. For each step, the biggest value under the pooling filter is selected to the output matrix of the pooling operation. [19]

After a number of preprocessing layers, the input tensor is flattened and it's passed to the dense layer. The dense layer operates similarly to the MLP discussed in the previous chapter. The prediction made by the CNN is determined by the output of the dense layer. [19]

3.4.5 Metrics

In this thesis, model performance is examined with the Receiving Operating Characteristic (ROC) and the Area Under Curve (AUC). Hanley and McNeil reviewed ROC and AUC in their 1982 article "The meaning and use of the area under a receiver operating characteristic (ROC) curve" [20]. ROC and AUC are used to evaluate the separation ability of a classifier in binary detection tasks. The ROC curve visualizes how changing the prediction threshold affects the True-Positive Rate (TPR) and the False-Positive Rate (FPR). Figure 3.7 portrays an example of a ROC-curve. TPR and FPR are defined as

$$TPR = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

$$TPR = \frac{\text{False Positives}}{\text{False Positives} + \text{True Negatives}} \tag{3.9}$$

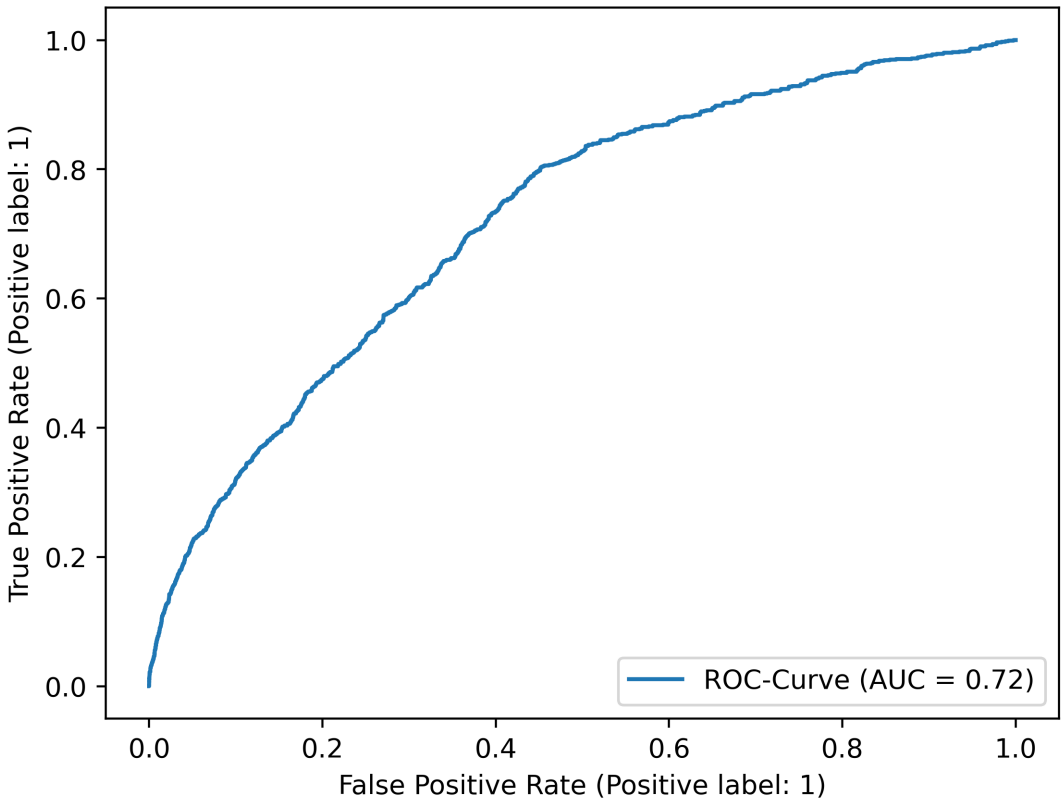

Figure 3.7. *An example of an ROC curve.*

TPR and FPR values are calculated for each probability threshold that the model's output is tested on. The resulting values are plotted to a figure with the TPR on the y-axis and the FPR on the x-axis. AUC measures the area between the x-axis and the ROC curve. The perfect model would result in a figure where the TPR reaches 1 vertically when the FPR is at 0. The AUC of the perfect ROC is 1.0.

ROC and AUC are chosen as the performance metrics due to being informative in highly imbalanced class distributions. In cases where one class vastly outnumbers the other, traditional metrics like accuracy become misleading. Models might reach near-perfect accuracies while not learning to differentiate between classes at all.

4. DATASET

This chapter explores the data used to build the OPSO model. The dataset consists of event data collected from the Finnish professional ice-hockey league Liiga with Wisehockey's tracking system. In addition to events, all necessary metadata, such as rink configurations for different rinks, was also gathered from Wisehockey. Full tracking data was also available for all matches.

4.1 Data features

The dataset used to build the model consists of 250 matches from the 2022/2023 Liiga season. All shot events and successful pass events were gathered from those matches.

Each sample in the dataset contains the following features:

1. Event id
2. Calculation id
3. Timestamp
4. Controlling player id
5. Puck position
6. Controlling team
7. Active player positions
8. Active player velocities
9. Active player teams
10. Active player roles
11. Direction of play
12. Left goal line
13. Left blueline
14. Right blueline
15. Right goal line
16. Label (Goal/No goal)

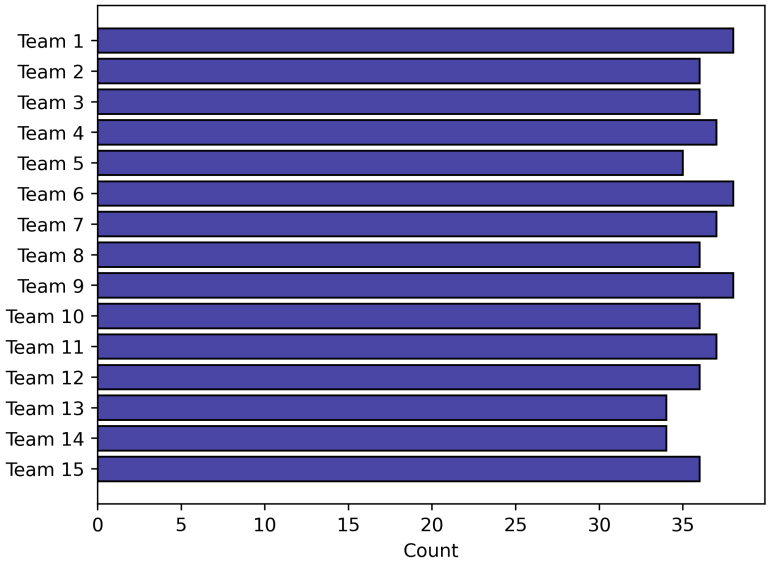

Figure 4.1. *The number of matches played by each team in the dataset.*

Event type	Positive samples	Negative samples	Total
Pass event	1 605	140 837	142 442
Shot event	1 593	24 112	25 705
Total	3 198	164 949	168 147

Table 4.1. *The distribution of positive and negative samples in the dataset.*

The label "Goal" or "No goal" is determined by the controlling team scoring within 5 seconds from the event. For shot events, this means that the team can also score from possible chances following the shot, not only on the shot in question. To minimize the effect of teams' differences in play styles, the dataset was collected from all teams in Liiga. The number of matches was determined by the progression of the season up to the point of collecting the dataset.

Figure 4.1 presents the number of matches for each Liiga team in this dataset which shows that there is an even distribution of matches featuring each team. Table 4.1 contains the number of events in the dataset. The dataset contains considerably more negative than positive samples. This is due to the inclusion of pass events in the data. The percentage of passes leading to a goal within 5 seconds is 1.1% whereas 6.2% of shots in the dataset lead to a goal. The difference is intuitive as shots are always attempts to score whereas the puck is passed between teammates for various reasons. Figure 4.2 visualizes the discrepancy between positive and negative samples.

Figure 4.3 visualizes the locations of events leading to goals in the dataset. The figure highlights that the area in front of the goal where controlled on-puck events result in

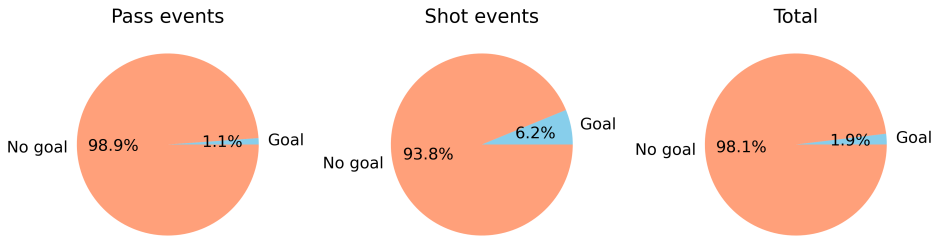

Figure 4.2. *The percentage of positive and negative samples in the dataset.*

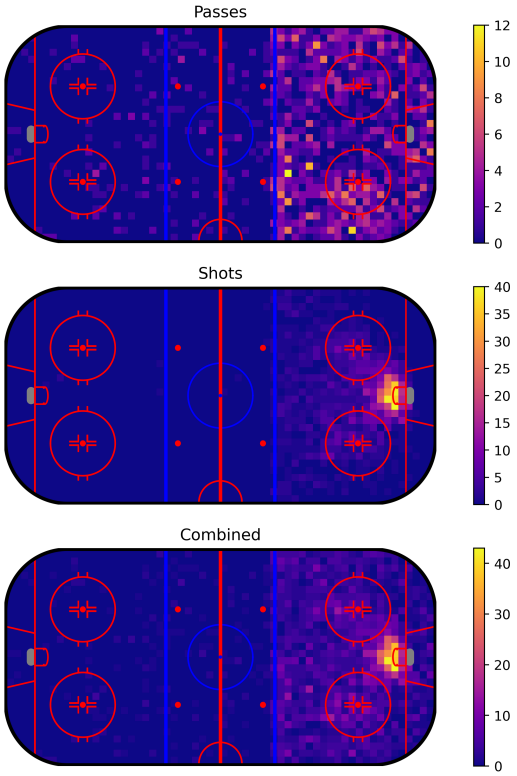

Figure 4.3. *The positive sample locations in the dataset normalized to the standard rink size and towards the right goal.*

scoring goals most often. Passes leading to goals are spread out around the offensive zone. This is expected since passes can be directed toward the front of the goal from any position within the offensive zone. Shots that result in goals are concentrated in the area directly in front of the goal.

Dimension	Location from center
Left end board	-30
Left goal line	-26
Left blue line	-7.3
Right blue line	7.3
Right goal line	26
Right end board	30
Top side board	15
Bottom side board	-15

Table 4.2. *The dimensions of the standard rink.*

4.2 Preprocessing

Different sections of this thesis required varying levels of preprocessing for the data. Before the event data could be used to develop the passing- and scoring probability models in Chapter 5, it needed to be normalized. First, the samples were normalized such that in each instance the controlling attempted to score toward the goal on the right side of the rink. The samples were subsequently normalized to a standardized rink size since the rinks in Liiga aren't identical. Table 4.2 contains the dimensions of the standard rink. The process of normalizing coordinates to the standard rink size renders velocities incomparable. Therefore, the fully normalized data is utilized solely in contexts where player velocities are not required.

A directionally normalized dataset was also produced for the CNN testing conducted in Chapter 6. The data is normalized so that the attacking team is always trying to score on the goal on the right side of the image. The raw data is used with no normalization in the match analysis section of Chapter 6 as well as in all other OPSO output figures presented in this thesis. Directional normalization is not used in these to avoid having mirrored images between camera footage and OPSO figures.

In Wisehockey's coordinate system, location (0,0) is at the center of the rink, the x-coordinate increases towards the right side of the rink, and the y-coordinate increases towards the top of the rink. The same coordinate system is used in all of the rink figures in this thesis. The dataset was divided randomly into training and testing sets. The same split is used throughout the thesis. The division was conducted on match level, meaning that all events from one match were assigned to training or testing data. The training data contained 200 matches and the remaining 50 matches were used for testing.

5. OFF-PUCK SCORING OPPORTUNITY DETECTION

The aim of this thesis is to develop a model to predict the probability of scoring within 5 seconds from the next on-puck event for the controlling team from any location within the rink. The 5-second prediction length was selected experimentally by increasing the prediction period until the model performance declined. The instantaneous state of the game is divided into three components.

1. Pitch control probability: the probability that location *p* is controlled by the team currently controlling the puck. Denoted by I_p.
2. The probability of passing: the probability that the puck is successfully moved to location *p* for the next on-puck event. Denoted by T_p.
3. The probability of scoring: the probability of scoring within 5 seconds following an event from location *p*. Denoted by S_p.

The total probability of scoring within 5 seconds from the next on-puck event for the controlling team is defined as

$$P(S|H) = \sum_{p \in \mathbb{R} \times \mathbb{R}} P(S_p \wedge Tp \wedge I_p|H) \tag{5.1}$$

H represents the instantaneous state of the game, containing the positions and velocities of the players and the puck. The following chapters will address each component of Equation (5.1). [2, 3]

5.1 Pitch control probability

The purpose of pitch control is to measure the probability that location *p* is controlled by the team currently controlling the puck. The implementation follows the parametric method presented by Fernandez et. al in their 2018 paper "Wide open spaces" [3]. In their paper, they present the idea of continuous ownership of space within the playing field instead of each point belonging only to one of the teams. This idea is well-founded, taking into account the prevalence of multiple players contesting for key areas in ball sports. Each team's influence is summarized by the influence of individual players. This results in a smooth surface of control for both teams. These are used to determine the

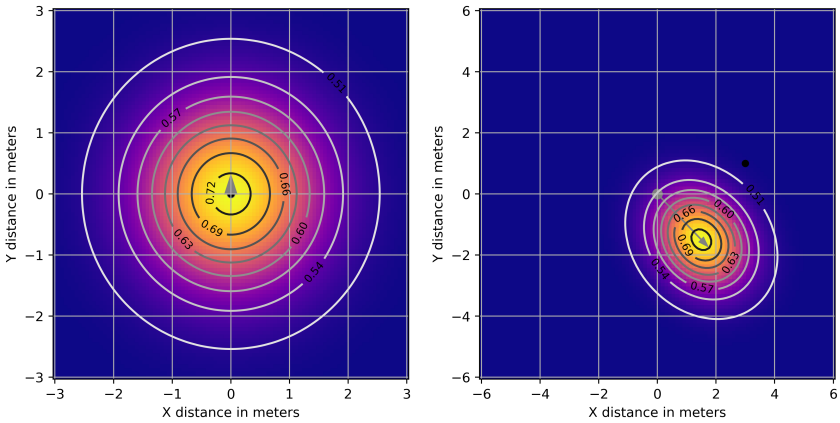

Figure 5.1. *An example of player influence. The left figure illustrates the influence of a player standing still with the puck. The right figure depicts the influence of a player moving at 4.24 m/s while being 3.16 m away from the puck. Adapted from source [3].*

pitch control probability for every location on the playing surface.

The influence of an individual player takes into account the player's velocity, location, and their distance to the puck. The player can only control areas they are close to, so taking their location into account is intuitive. The player's velocity and direction of movement also affect their area of influence. A player moving at a high speed is more likely to be able to control areas that they are moving towards. Consequentially, a moving player is also less likely to control areas they are moving away from. The player's distance to the puck is also considered to affect the area of influence a player has. The further away the player is from the puck, the larger area they are considered to influence. A player near the puck has less time to control the puck if the puck moves away from its current position. Conversely, a player further from the puck can recover the puck from a larger area if the puck is moved away from its current position. Using these three parameters, Fernandez defines the influence *I* of player *i* at a location *p* at a point in time *t* with a multivariate normal distribution with mean $\mu_i(t)$ and covariance matrix $\sum_i(t)$, given the player's velocity $\vec{s}$ and angle θ. [3]

Players' influence likelihood is defined as

$$I_i(p,t) = \frac{f_i(p,t)}{f_i(p_i(t),t)} \tag{5.2}$$

where the influence of player *i* is defined by a standard multivariate normal distribution which is normalized by the value of *f* at the player's current location $p_i(t)$. The player's velocity and distance to the puck are taken into account when adjusting the multivariate normal distribution. Players influence function is defined by Fernandez as

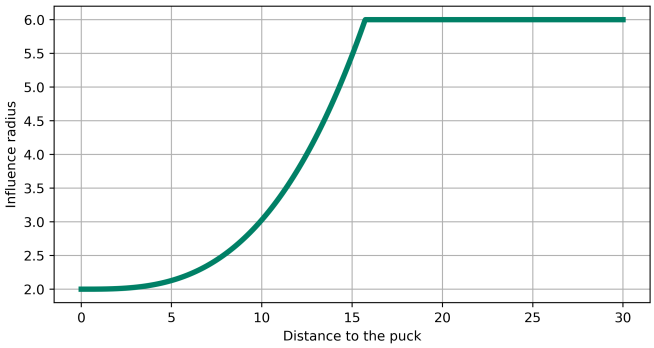

Figure 5.2. *The effect of the player's distance to the puck on the influence radius of the player. Adapted from source [3].*

$$f_i(p,t) = \frac{1}{\sqrt{(2\pi)^2 det\Sigma_{i,t})}} exp - \frac{1}{2}(p - \mu_i(\vec{s_i}(t)))^\intercal \Sigma_{i,t}^{-1}(p - \mu_i(t)) \tag{5.3}$$

The covariance matrix is adjusted to factor in movement speed and location to the player's influence function. The covariance matrix can be expressed as a function of its eigenvectors and eigenvalues through singular value decomposition in Equation (5.4) where *V* is the matrix whose columns are the eigenvectors of Σ and *L* is the diagonal matrix whose non-zero values are the corresponding eigenvalues. [3]

$$\Sigma = VLV^{-1} \tag{5.4}$$

Let $R = L$ and $S = \sqrt{L}$ in order to define *R* as the rotation matrix and *S* as a scaling matrix. This allows the covariance to be expressed as

$$\Sigma = RSSR^{-1} \tag{5.5}$$

Equation (5.5) allows the direction and velocity of the player to be taken into account for the shape of the multivariate normal distribution. Rotation matrix *R* is defined as

$$R = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \tag{5.6}$$

where θ is the angle of the player's movement direction in relation to the goal line's normal. The player's velocity is defined by their speed vector $[s_x, s_y]$, and the scaling matrix *S* is defined in Equation (5.7). [3]

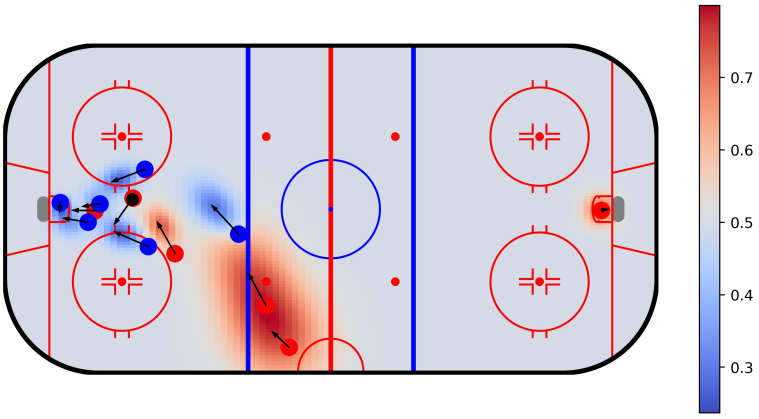

Figure 5.3. *An example of the pitch control model's output in a match setting.*

$$S = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix} \tag{5.7}$$

The player's relative speed and distance to the puck are taken into account in scaling the covariance matrix. Relative speed is defined in Equation (5.8) where the square of the player's speed is divided by the square of the theoretical maximum velocity which is set to $13\frac{m}{s}$. [3]

$$Srat_i(s) = \frac{s^2}{13^2} \tag{5.8}$$

The relationship between the player's distance to the puck and the player's influence radius is specified by Equation (5.9), where *d* represents the distance between the puck and the player. The minimum radius is set to 2 m and at most, the player can be considered to influence an area with a 6 m radius. The output of Equation (5.9) is visualized in Figure 5.2. [3]

$$R_i = \min\left(radius_{min} + \frac{d^3}{\frac{18^3}{6}}, radius_{max}\right) \tag{5.9}$$

Scaling matrix $S_i(t)$ is expanded in the x-direction and contracted in the y-direction by the player's relative speed defined in Equation (5.8).

$$S_i(t) = \begin{bmatrix} \frac{R_i(t)+(R_i(t)Srat_i(\vec{s_i}(t)))}{2} & 0 \\ 0 & \frac{R_i(t)-(R_i(t)Srat_i(\vec{s_i}(t)))}{2} \end{bmatrix} \tag{5.10}$$

Finally, the covariance matrix is defined in Equation (5.11). [3]

$$COV_i(t) = R(\theta, t)S_i(t)S_i(t)R(\theta_i(t), t)^{-1} \tag{5.11}$$

The multivariate distribution's mean $\mu_i(t)$ is placed at half the magnitude of the speed vector $\vec{s}$ in Equation (5.12).

$$\mu_i(t) = p_i(t) + \vec{s_i}(t) \cdot 0.5 \tag{5.12}$$

The pitch control model is defined in Equation (5.13) where the influences of individual players are aggregated teamwise before the subtraction is transformed through the logistic function to obtain the total pitch control for point p at time t within the range $[0, 1]$. Terms i and j represent the players on opposing teams. [3]

$$PC(p, t) = \sigma(\sum_i I(p, t) - \sum_j I(p, t)) \tag{5.13}$$

Figure 5.3 depicts the pitch control in a match. For OPSO, the player in control of the puck doesn't contribute to the pitch control. The reasoning for this is that their influence on the attacking effort is taken into account when the puck is played to them in the previous event.

5.2 The probability of passing

The second component of the model is the probability density of the controlling team successfully transitioning the puck to point *p* in the rink [2]. This component is represented by a multivariate normal distribution mimicking the displacement of the puck between consecutive events. The events include both shot- and pass events.

The displacements of the puck between consecutive events are presented in Figure 5.4. The lower graph presents the multivariate normal distribution modeling the displacement of the puck. The puck is placed in the middle and the axes represent the distance from the puck. As Spearman mentions [2], we can assume that players aren't acting randomly but instead trying to make successful passes to players who are likely to receive the pass. Previously introduced pitch control probability can be used with the distribution of the puck displacement to produce the probability of transitioning the puck. The transition probability function is presented in Equation (5.14) where $PC(p, t)$ represents the pitch control probability function.

$$PT(PC(p, t), t) = \mathcal{N}(\mu, \Sigma) \cdot PC(p, t) \tag{5.14}$$

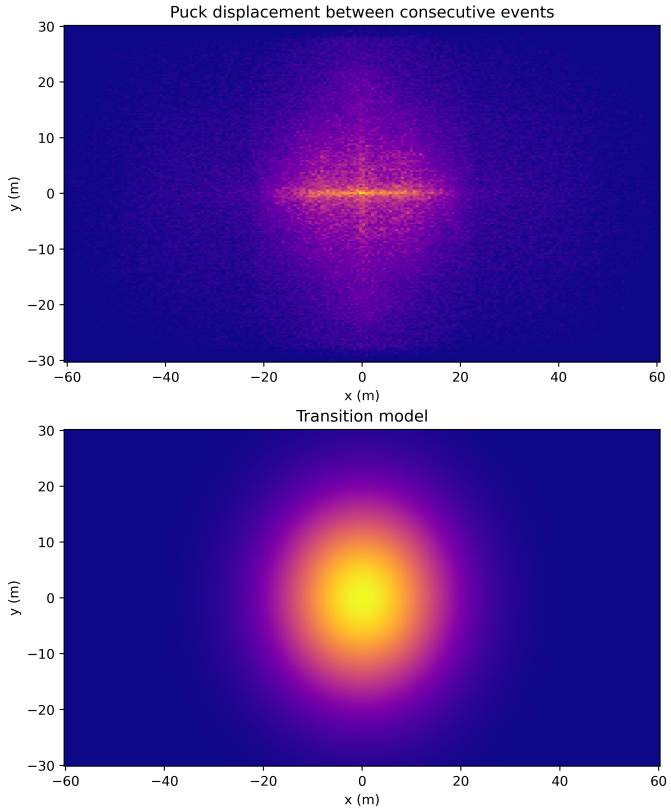

Figure 5.4. *The displacement of the puck between consecutive events and the transition model.*

Variable	Variance
Δx	346.97
Δy	146.01

Table 5.1. *Direction-wise variance between consecutive events.*

The covariance matrix for the multivariate normal distribution was formed by calculating the distances between consecutive events in the dataset. Table 5.1 shows the variances between consecutive events for Δx and Δy. At first, the covariance matrix was formed directly from these variances. However, experimentation with different covariance matrices revealed that a symmetric distribution, achieved by symmetric diagonal values in the covariance matrix, yielded slightly better results for the full model. For this reason, the final version of the transition model was produced with a covariance matrix where the covariance was determined by Δy shown in Equation (5.15).

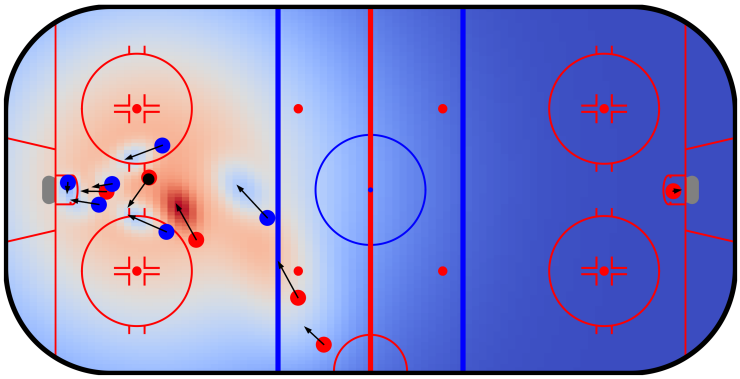

Figure 5.5. *An example of the passing probability model's output in a match setting.*

$$\Sigma = \begin{bmatrix} 146 & 0 \\ 0 & 146 \end{bmatrix} \tag{5.15}$$

Figure 5.5 depicts the output of the passing probability model which combines the pitch control model with the puck transition probability density.

5.3 The probability of scoring

The third component of the model is the probability of scoring within 5 seconds after a controlled event from location *p*. Events are limited to pass- and shot events. Three different methods were proposed to determine the scoring probability: logistic regression, exponential distribution, and MLP. The training data was represented differently for each model to maximize the models' performance. The task given to each model was the same: does the team score within 5 seconds after an event at location *p*?

First, for the exponential distribution, the positive samples were collected from the training dataset. The locations of the events were represented as distances from the goal. An exponential distribution was fit to the distances. For logistic regression, the representation of the data was set to four features:

1. Distance from goal $\in [0, \inf)$
2. Angle to the goal line $\in [0, 180]$
3. In the offensive zone $\in \{0, 1\}$
4. Behind the goal $\in \{0, 1\}$.

The last two features were Boolean values giving additional information about the position of the event within the rink. The event locations were converted to polar coordinates

to direct the highest scoring probability region toward the goal. For the third and final scoring probability model proposal, the MLP was trained on events with the following representation:

1. Event x-coordinate $\in [-30, 30]$
2. Event y-coordinate $\in [-15, 15]$
3. In the offensive zone $\in \{0, 1\}$
4. Behind the goal $\in \{0, 1\}$.

The two latter feature flags are used with the same reasoning as in logistic regression. The event location was represented as raw coordinates rather than as distance and angle because coordinate representation led to a smoother scoring probability map with the MLP.

The following structure was selected for the MLP model:

- Input layer
- 3 hidden layers with:
  - 50 nodes
  - 100 nodes
  - 10 nodes
- Output layer

ReLu-activation was selected for the activation function.

The resulting scoring probability maps are displayed in Figure 5.6. The scoring probability maps were produced by querying each model for the predicted scoring probability for each location within the rink. The ROC curves and AUC scores of the scoring probability model candidates are presented in Figure 5.7. The test dataset was used to calculate the ROC curve. The figure shows that the MLP model provides the highest performance of these three models. The scoring probability maps derived from the logistic regression and the exponential distribution are almost identical, the only difference being the higher probabilities of scoring produced by the logistic regression. As linear models, they have limited ability to represent the varying values of scoring locations with more complexity. The feature flags given to the logistic regression aren't visible in the output of the model as the model can't contain a cutoff for the values of locations behind the net or outside the offensive zone. While it could be argued that the blueline doesn't inherently affect the scoring probability of an event, it does influence the state of the game, which is heavily impacted by the offside rule. For instance, there can not be attackers screening the opposition goalkeeper for an event that takes place outside the offensive zone due to the offside rule. Events outside the offensive zone are likely to lead to scoring only in

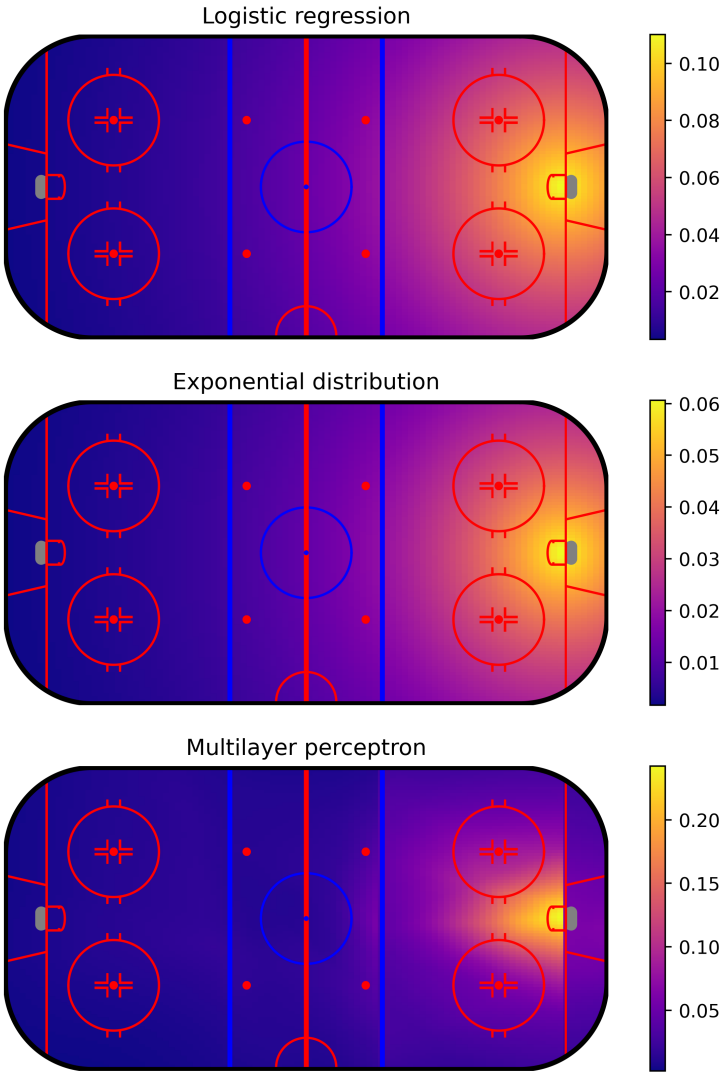

Figure 5.6. *The scoring probability maps produced by logistic regression, exponential distribution, and multilayer perceptron.*

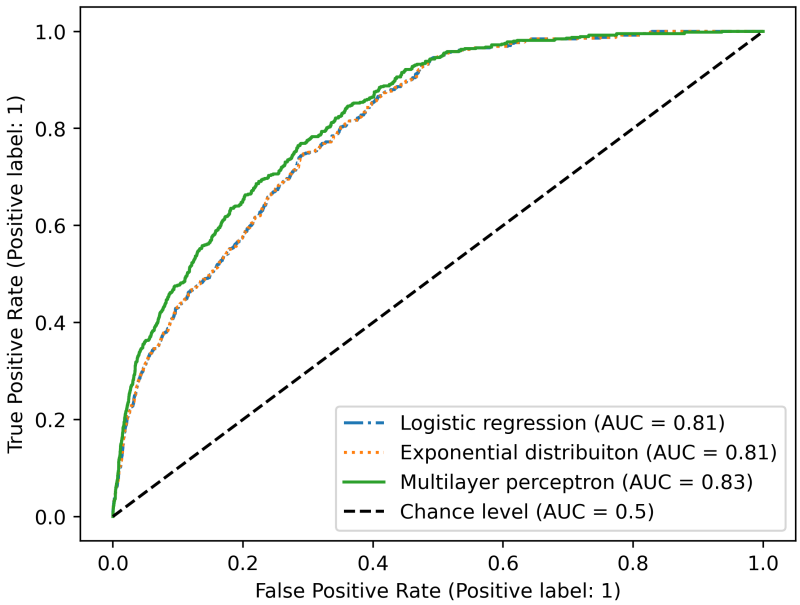

Figure 5.7. *The scoring probability model candidates' ROC curves and AUC scores for predicting goal-scoring within 5 seconds.*

direct attacks as the offside rule sets boundaries on how the team must proceed into the offensive zone.

The MLP-based scoring probability map values areas non-linearly. The area in front of the goal is the most valuable in the offensive zone. Events from tighter angles receive high values if they are near the goal line. A clear cutoff happens when crossing the goal line or the offensive blueline. However, a closer inspection of the MLP model shows that there is a slight bias towards the left side of the rink. While it is possible that the left-wingers were exceptionally efficient with their goal-scoring output during the 2022/2023 Liiga season, it isn't desired to predict a higher probability of scoring if the event occurs on the left side of the net. The left-sided bias could be the result of overfitting and so the MLP model is not used as the scoring probability model. While it is included in the experiments in Chapter 6, the applications presented later excluded the model for this reason. The scoring probability model used in the applications is the exponential distribution.

5.4 Combined model

After building each component, the final model is ready for completion. The probability models are combined as in Equation (5.1) to produce the probability density of scoring within 5 seconds of the next on-puck event from any location p within the rink given the current state of the game. As the model's output is a probability density map, the output must be spatially integrated to produce the probability of the controlling team scoring. An

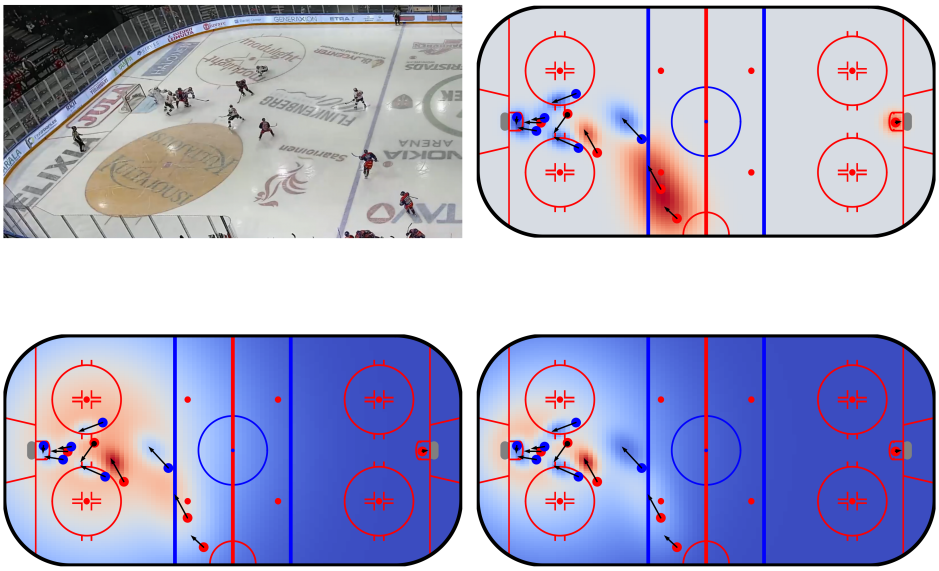

Figure 5.8. *An example of how the components form the OPSO output in a match setting. The top left image shows the scenario from the camera feed. The top right figure portrays the pitch control model. The bottom left figure adds the transition model to the pitch control model. The bottom right figure applies the scoring probability model on the transition and pitch control models to complete the OPSO output for this scenario.*

example of the model's output is presented in Figure 5.8. The home team's players are represented by red scatter points and the away team's players by blue scatter points. The arrows on the players show their movement speed and direction. Each arrow points at where the player is moving 0.5 seconds from now at their current speed and movement direction. In this example, the home team player with the puck moving towards the center from the right wing is about to take a backhanded shot at goal. OPSO suggests that the home team player arriving from the left wing has the highest probability of scoring from between the faceoff circles. The area is highlighted as the defenders near the puck are moving towards the goal, leaving the area to be controlled by the arriving attacker.

6. EXPERIMENTS

This chapter presents the series of experiments conducted with the OPSO model. The first section proposes two validation methods for OPSO. The next section reviews how the model estimates the highest scoring probability regions during matches. The last section presents possible applications for OPSO.

6.1 Model validation

The OPSO models were tested with a dataset consisting of 50 Liiga matches. The task for the models was to predict whether the controlling team was going to score within 5 seconds of an event. The performance of the models was evaluated with ROC curves and AUC scores. Two separate validation methods were used with the same dataset to test the OPSO models. First, OPSO images were produced from the training and testing datasets. These images were used to train and test a CNN model to verify that the state of the game can be represented through the heatmap produced by OPSO without losing any information. Alongside the CNN validation, OPSO scores for the same events were used to measure the models' AUC scores. Both validation methods aimed to exceed the AUC scores set by the scoring probability models in Chapter 5.

6.1.1 CNN validation

Figure 6.1 depicts the architecture of the CNN model trained for the validation of OPSO's output. The selected architecture is a variant of Yann Lecun's LeNet architecture where

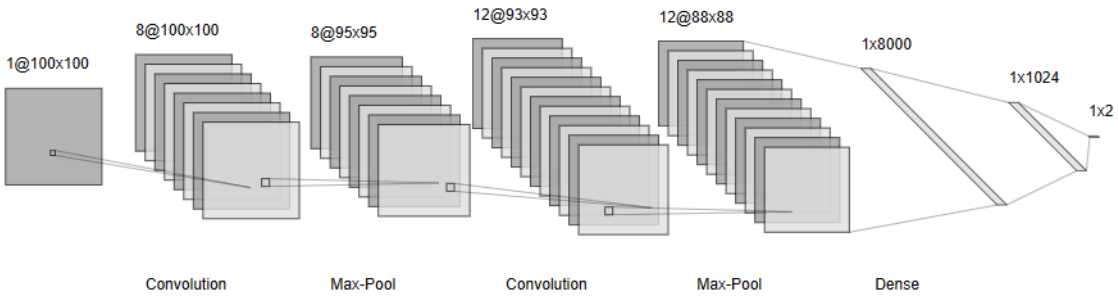

Figure 6.1. *The validation CNN architecture.*

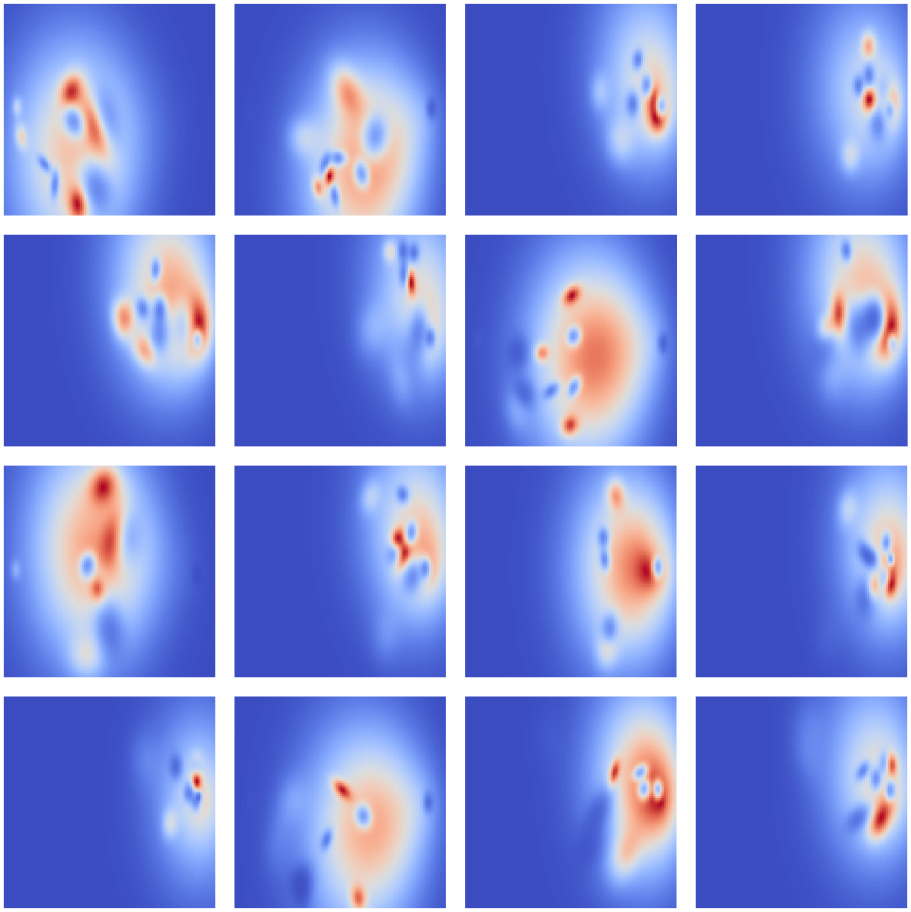

Figure 6.2. *Examples of positive samples of OPSO output. The controlling team scored a goal to the right side of the rink within 5 seconds of these events.*

convolutional layers are followed by subsampling layers before the signal is fed to a fully connected network [16, p. 100]. The model was trained with a training dataset of 200 matches. An OPSO heatmap was created for each event in the dataset. Another option would have been to create heatmaps for every second in the dataset. However, this would have resulted in a highly imbalanced dataset between positive and negative samples. Including samples from all teams in Liiga would have increased the dataset's scale drastically. Creating heatmaps only for events was chosen to include a larger variety of teams and goal-scoring situations in the dataset. The task for the CNN model was to predict whether the team was going to score within 5 seconds based on the heatmap. The labels for these heatmaps were the same as they were for the base scoring probability models presented in Chapter 5. Examples of the images are presented in Figures 6.2 and 6.3. The images are presented with a blue-white-red colormap to ease the perception for the human eye. For the CNN model, the images were presented as grayscale images. The grayscale images were normalized within $\in [0, 255]$.

The trained model was tested with the evaluation dataset consisting of events from 50

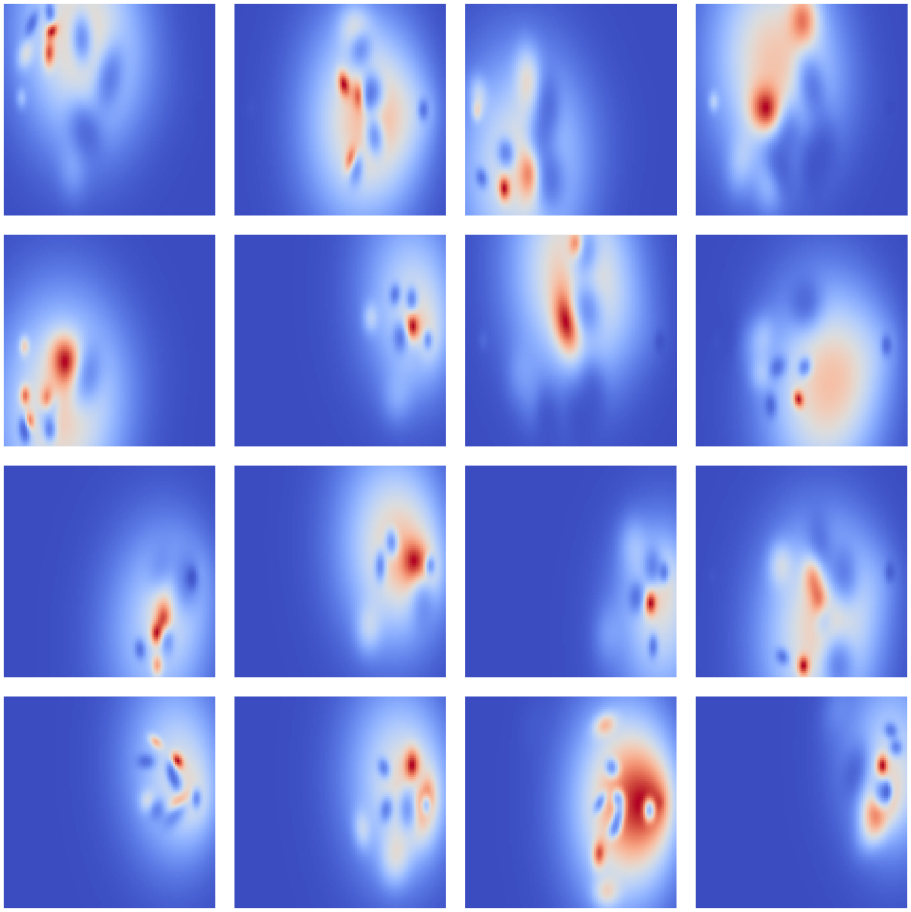

Figure 6.3. *Examples of negative samples of OPSO output. The controlling team did not score a goal within 5 seconds of these events.*

matches. Different layer configurations were experimented with, but the presented network structure was the best-performing model with an AUC score of 0.72. Even the best-performing CNN models weren't able to reach the AUC scores of the scoring probability models examined in Chapter 5. Given the lack of improvement to the AUC scores, the CNN model doesn't have any applications moving forward. However, the score of 0.72 does suggest that there is an informative signal in the OPSO heatmaps as the result is clearly above a random classifier.

It was expected that the CNN models might struggle to determine if the situation presented with the heatmap would lead to a goal given how small the differences are between positive and negative samples. In addition, the details in the input images are very broad as the pitch control algorithm results in blobs of control area for each team making it difficult to distinguish the difference between situations in games. Regardless, the CNN model does learn the difference between events leading to goals or not to a satisfactory degree.

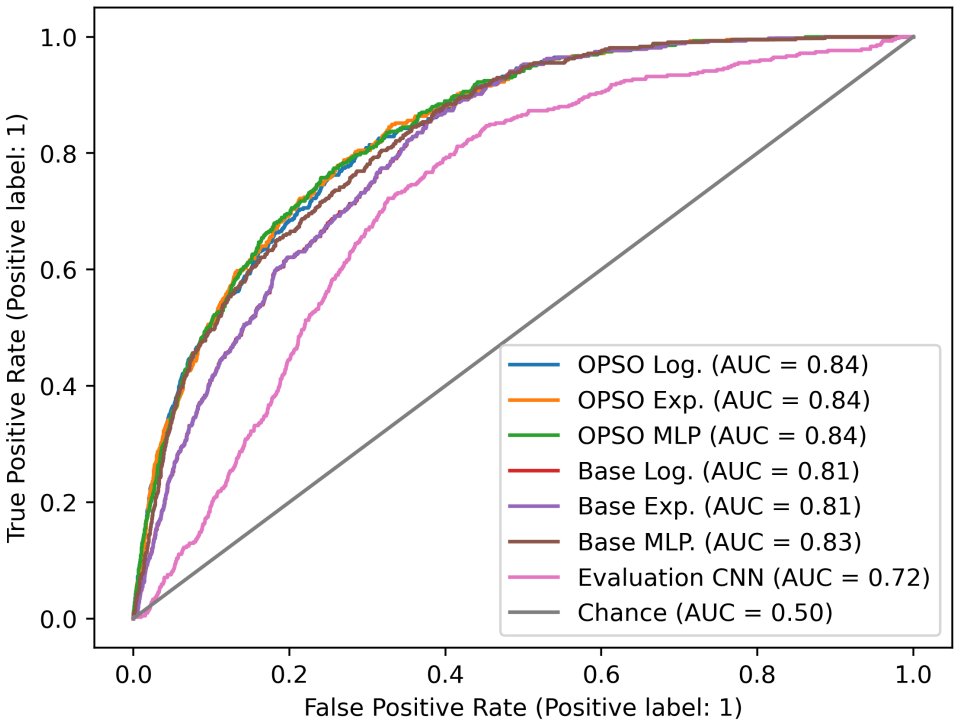

Figure 6.4. *ROC curves for the OPSO models, the base scoring probability models, and the validation CNN for predicting goal-scoring within 5 seconds.*

6.1.2 Model performance

The scoring probabilities were integrated from the OPSO heatmaps for the events in the testing dataset. The ROC curves and AUC scores are shown in Figure 6.4. Plotting the ROC curves and calculating the AUC scores show that in comparison to the scoring probability models, OPSO slightly improves the detection accuracy. For logistic regression and exponential distribution, the difference is a notable 0.03 while the MLP is only improved by 0.01 units. It's noteworthy that while the scoring probability models had different AUC scores, OPSO models built on them all reached the same AUC score. This can be interpreted as the ceiling for the detection ability of the model with these components. Further improvements are considered in Chapter 7. The results show that the OPSO method does provide an improved detection ability compared to the base scoring probability models which was the aim for this thesis.

6.2 Match analysis

OPSO's purpose is to predict where the attacking team is the most likely to score from given the current state of the game, in addition to predicting the probability of scoring. Evaluating the accuracy of the location of the highest scoring probability is challenging

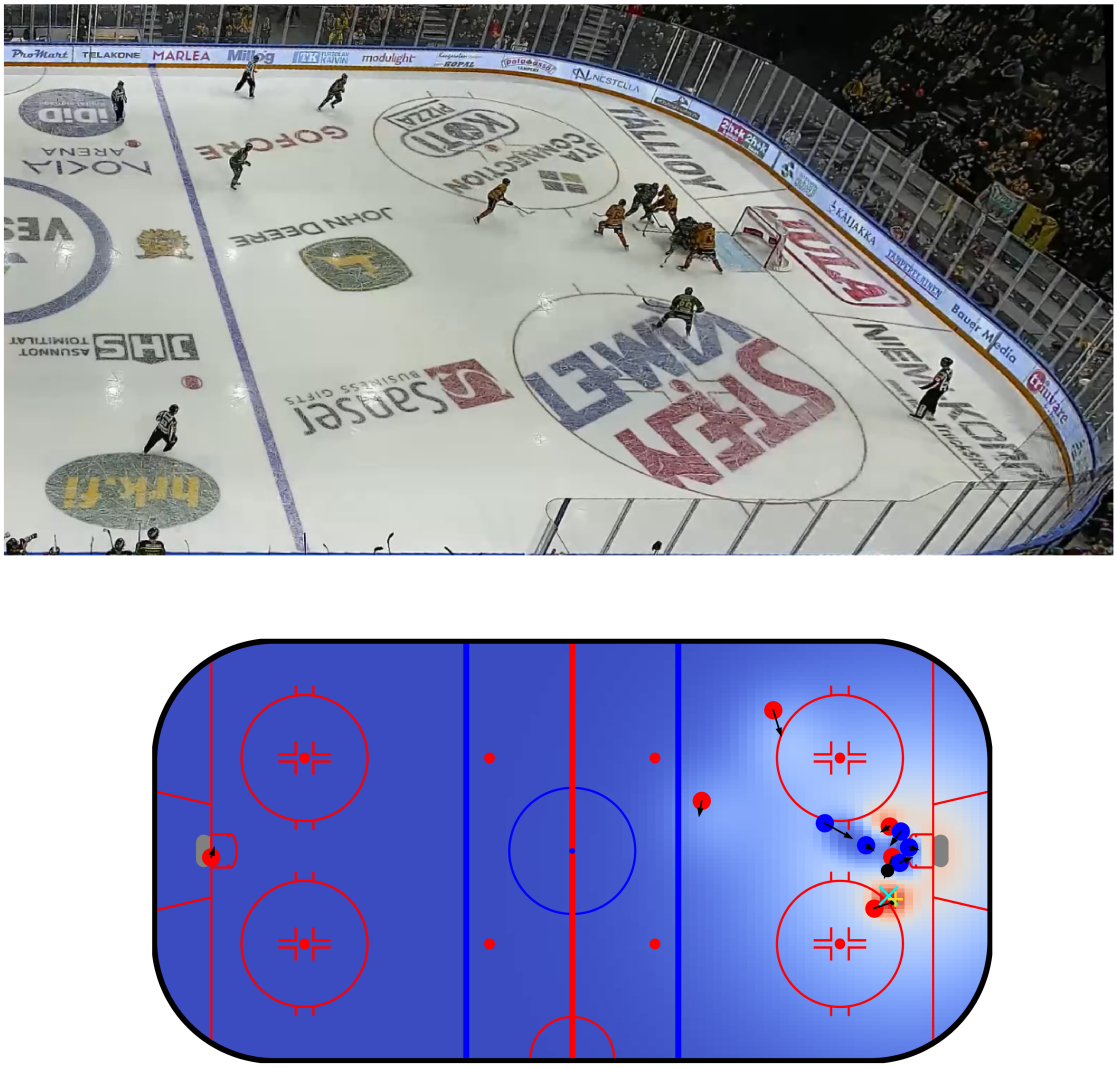

Figure 6.5. *An example of the OPSO model's output in a match setting. The turquoise marker points to the position where the attacking team scored the goal and the yellow marker points to where the OPSO model predicted that the attacking team was the likeliest to score from.*

with traditional evaluation metrics as there isn't a ground truth for cases where teams don't score even if they can create high-danger scoring opportunities. Examining passes leading to goals gives more insight into the model's ability to predict high-probability scoring areas.

Figure 6.5 shows a still image from a match and the corresponding OPSO figure. At the moment pictured, the attacking players in front of the net win the puck and pass it to the player waiting on the right wing. The OPSO output shown in the graph below shows that the predicted probability of scoring is the highest right in front of the player who goes on to score in the situation. The turquoise and yellow markers point to the locations where the goal was scored and where the probability of scoring was the highest.

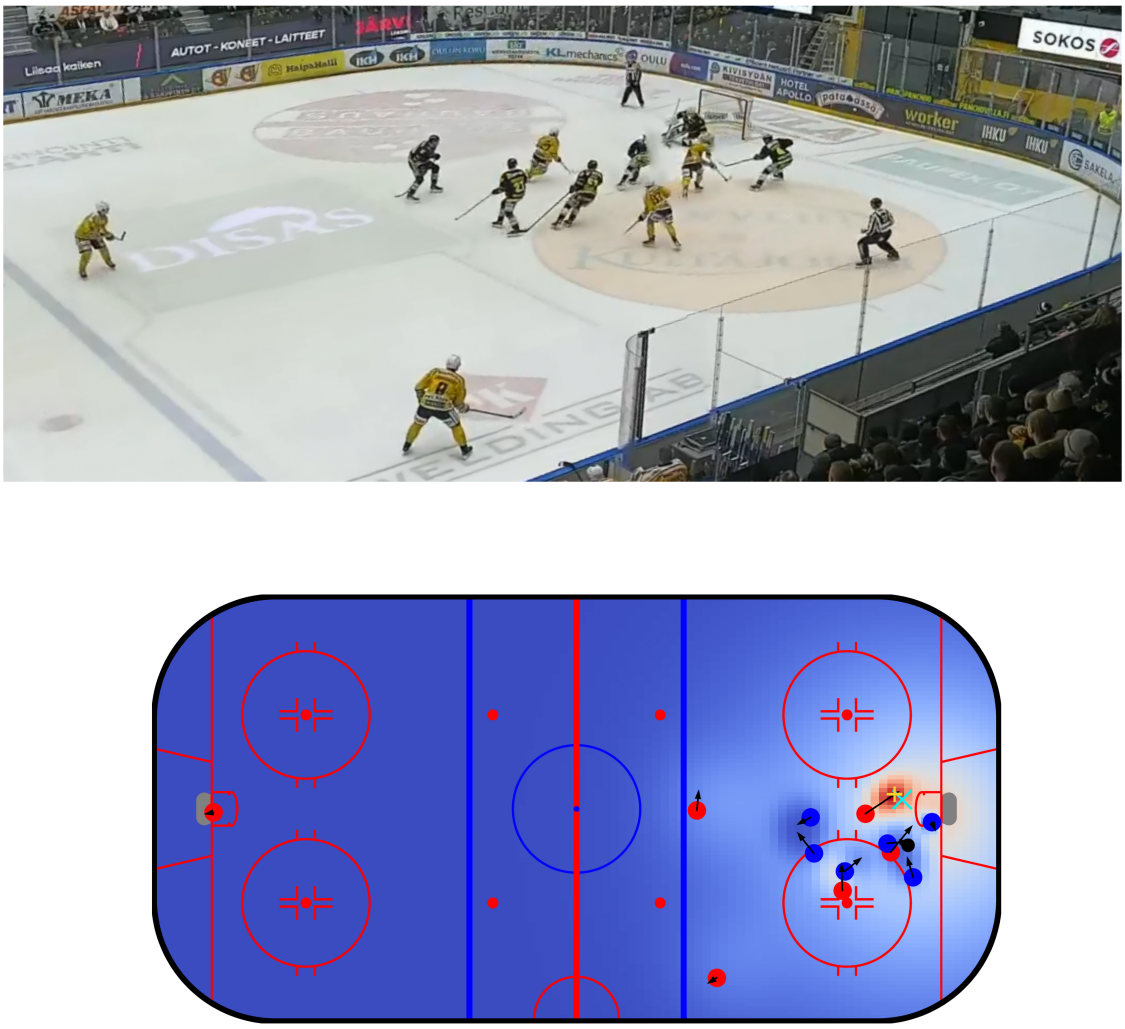

Figure 6.6. *Example 2 of the OPSO model's output in a match setting.*

Figure 6.6 presents another OPSO output from match play leading to a goal. In this situation, the team in yellow takes away the puck following an offensive zone faceoff which the defending team won at first. After the takeaway, two attackers break through on goal when the defending team is still in motion in the opposite direction. The image shows the moment that the player with the puck plays the lateral pass to the supporting attacker, who goes on to score. OPSO captures the high-danger scoring opportunity at the correct location.

Figure 6.7 shows the OPSO output for a scoring opportunity where a player approaches the goal from behind the net while their teammate carries the puck towards the opposite post. The defenders seem to have lost sight of the attacker nearing the goal. The attacker is noticed by their teammate who plays the pass leading to a goal. The lower graph shows that OPSO highlights the exact region where the goal-scoring opportunity is about

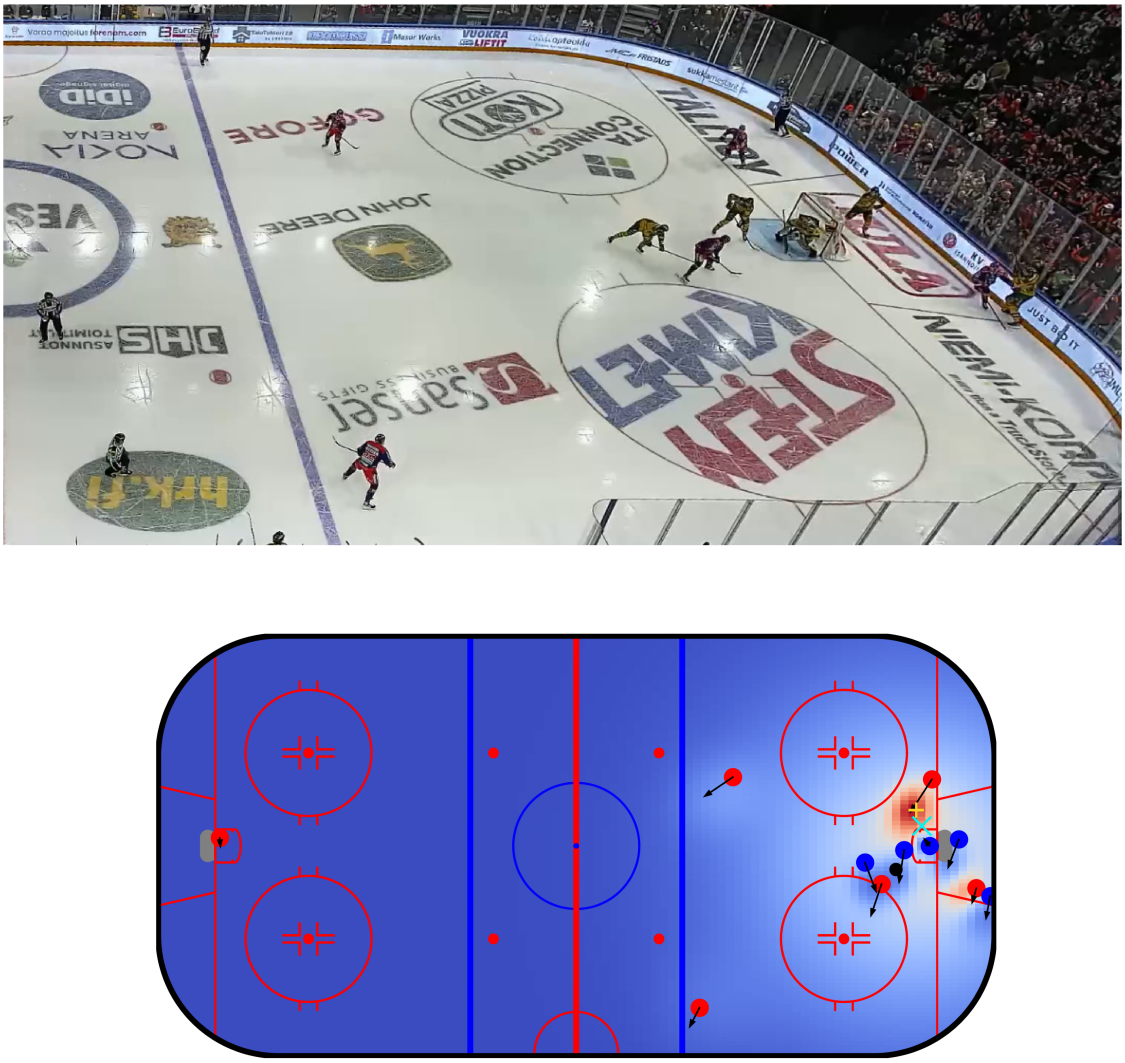

Figure 6.7. *Example 3 of the OPSO model's output in a match setting.*

to occur. Figures 6.5, 6.6 and 6.7 highlight how OPSO differentiates between having lots of space at the blueline and having a little space closer to goal. The attacking team's defenders positioned on the blueline aren't occupying spaces where goal-scoring is highly probable.

Figure 6.8 shows how the model performs on breakaways. In this situation, the attacking team has broken through on a 2-on-1 odd-man rush. The player controlling the puck has started to slow down and change direction to find a passing lane to the supporting player. This situation highlights how the pitch control algorithm adjusts the players' influence area according to their movement speed. In high-velocity play, the control areas push further ahead of the players and are narrower to the sides, which is intuitive for players moving at high speeds. They reach locations ahead of them likelier than regions to the side of their movement path. The OPSO output displayed on the bottom graph shows that the

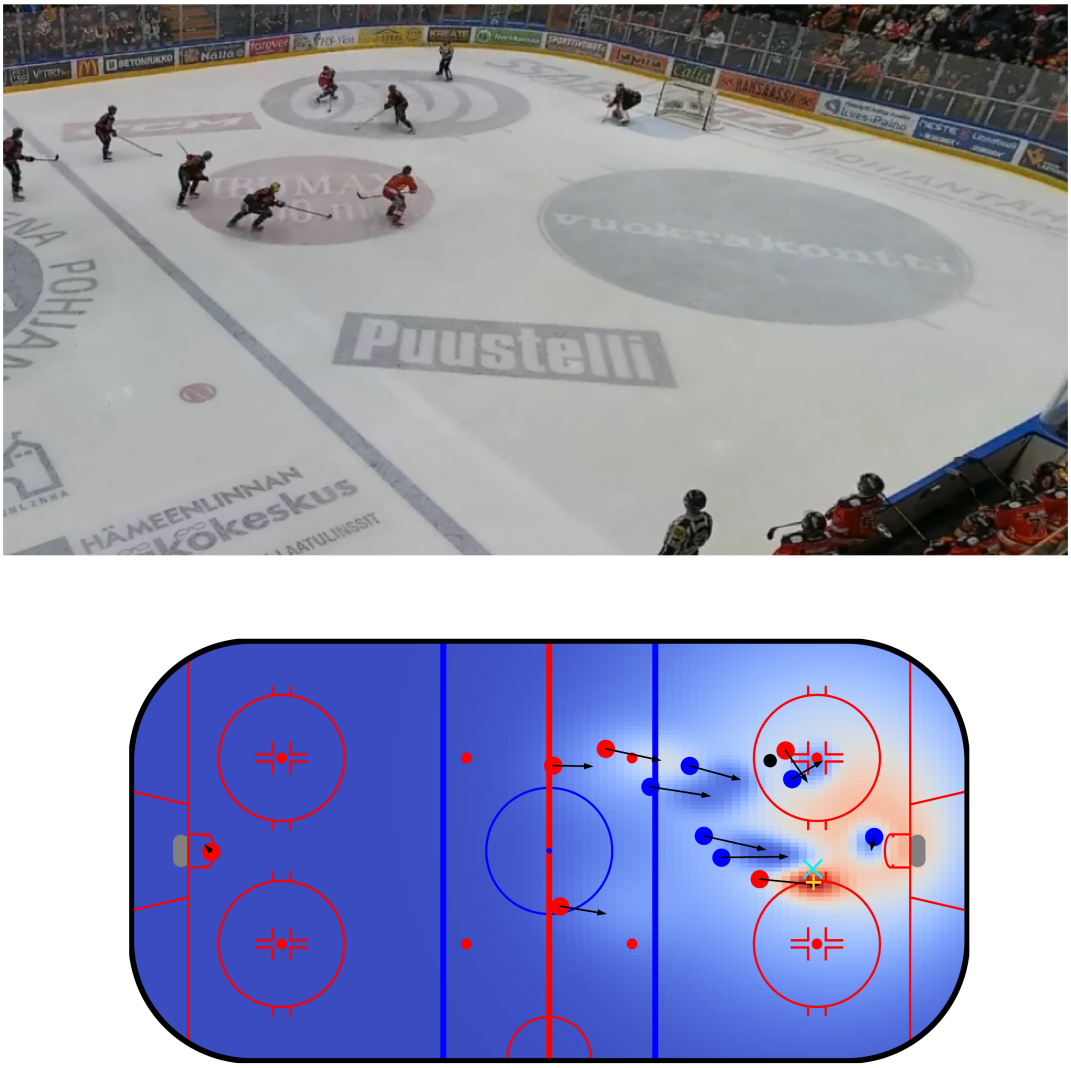

Figure 6.8. *Example 4 of the OPSO model's output in a match setting.*

supporting player's move toward the goal is recognized as the most valuable off-puck player.

Finally, Figure 6.9 depicts how OPSO performs in another higher velocity offense. While the player with the puck rushes between the faceoff circles, the supporting player glides backward toward a space near the right goalpost. The controlling player plays the pass and the supporting player scores the goal. While OPSO does again highlight the correct area of high-scoring potential, this case also shows the limitations of the model's current implementation. The player movement data doesn't contain information about the direction the players are facing. This results in situations like this where the player's influence area is behind them when they skate backward. In reality, the player skating backward can influence the area in front of them.

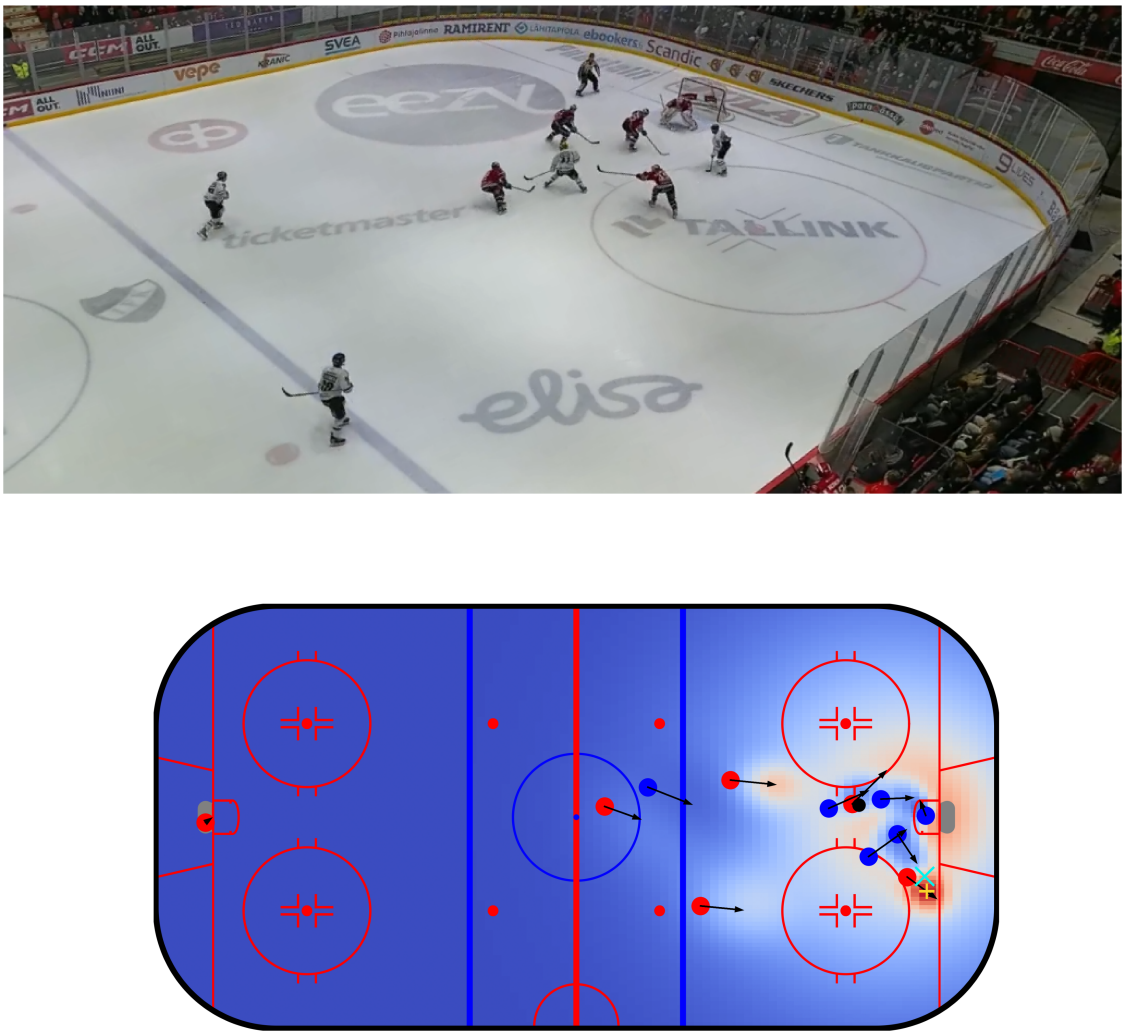

Figure 6.9. *Example 5 of the OPSO model's output in a match setting.*

6.3 Application examples

The ability to measure the attacking threat produced by an individual player or the entire team at any given moment within the playing field opens up new tools to analyze the progression of a match. Analyzing the threat produced by an individual player without the puck reveals where that player is likely to be found by their teammates to produce high-danger goal-scoring opportunities. Likewise, teams' tactical approach can also be studied by studying where the teams attempt to create spaces for scoring goals. The continuous measurement of OPSO can also be used to analyze the momentum of the match which can be used to easily visualize how the match progressed and how much each team was able to produce a goal-scoring threat at the opponent's end.

OPSO can be used to analyze which players position themselves in the best scoring op-

portunities. Furthermore, collecting OPSO for individual players reveals where they find the most valuable spaces to produce offensive threats. For some players, the locations are as one would expect given the player's role in the team. For others, OPSO shows how they are moving out of their expected positions to gain an advantage on the offense. Gathering OPSO scores from the areas influenced by players can also help identify quality players even if the opportunity to score never actually arrives. Figure 6.10 presents regular heatmaps and heatmaps sampled from OPSO next to them for five players during a single match. The individual player heatmaps were gathered from OPSO heatmaps by sampling OPSO from the players' influence areas at every moment. Comparing the heatmaps shows how OPSO can be utilized in analyzing where players seek scoring opportunities without the puck. The regular heatmaps cannot highlight scoring opportunities as clearly. The difference between regular heatmaps and OPSO heatmaps is the clearest on the second and fourth row in Figure 6.10. The OPSO heatmaps show that those players have positioned themselves in scoring positions at locations they did not otherwise occupy.

Momentum is an abstract measure of dominance in matches. In sports entertainment coverage, the momentum of the game is often described by a graph moving up and down according to which team was more in control of the match at which point. The progression of the match can also be represented through OPSO. Figure 6.11 shows the scoring probabilities integrated from OPSO for both teams collected during a single match for every moment either team has the puck. A window of 1000 frames is used to calculate the moving average OPSO scores to result in a smoother graph. The lower graph in Figure 6.11 presents the momentum graph calculated from the moving average scores. The momentum is calculated by subtracting the teams' OPSO scores. For this example, the away team's OPSO scores are given negative values to present the scores on the opposite sides of the x-axis.

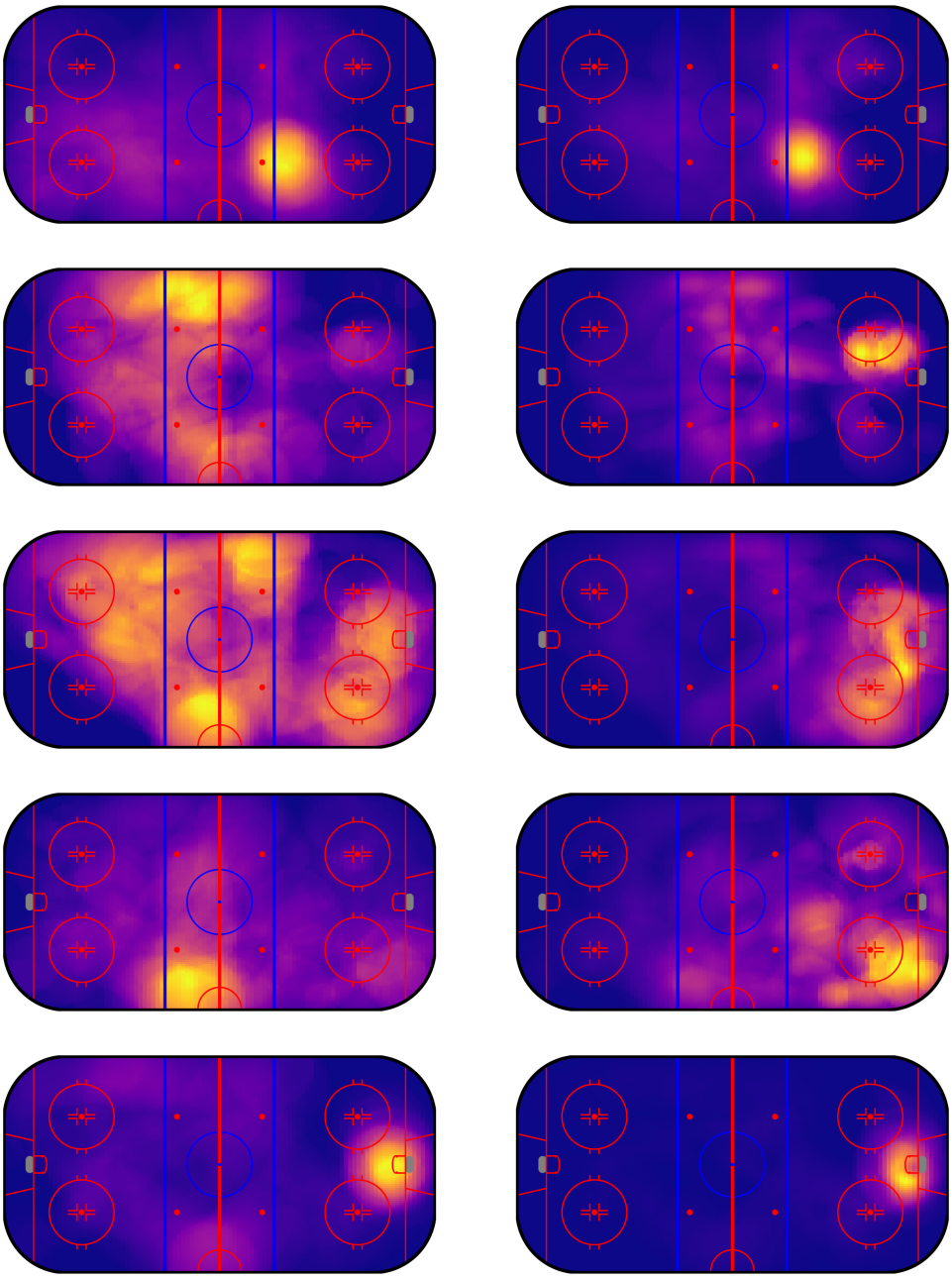

Figure 6.10. *Player heatmaps collected from one match when the player is on the ice and their teammates are in control of the puck. The figures on the left are collected as standard heatmaps and the figures on the right are sampled from OPSO.*

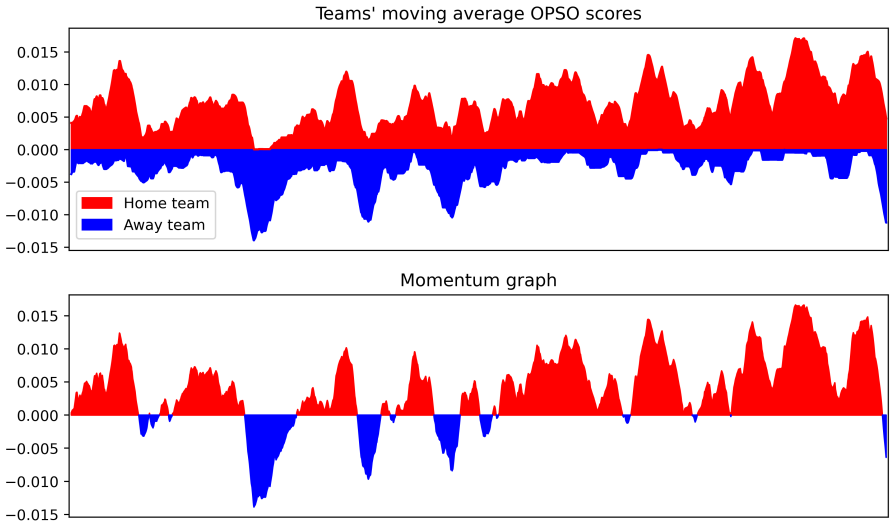

Figure 6.11. *Moving average OPSO scores for both teams and the momentum graph calculated from the moving averages.*

7. CONCLUSION

This thesis aimed to create a method that can be used to detect off-puck scoring opportunities in ice hockey. Tracking data from 250 professional hockey matches collected by Wisehockey was used to build the OPSO model, which is used to calculate the probability of scoring within 5 seconds of the next on-puck event from any location within the rink for the team in control of the puck.

7.1 Summary

The OPSO model was built of three individual components that represent different aspects of the game. First, the pitch control method measures the probability of teams' control of the playing surface. Second, the pass probability density function models the probability of the controlling team successfully moving the puck to a new location from its current location. Finally, a map of scoring probabilities is added to complete the model. The OPSO model produces a probability density function for scoring within 5 seconds of the next on-puck event from any location within the rink given the state of the game.

The ability to detect off-puck scoring opportunities opens up a variety of applications to present the progression of a match and to gather information during phases of games where no measurements were applicable beforehand. It provides a tool to track the value created by players' movements which can be utilized for scouting players, preparing for matches, or coaching purposes, when training attacking plays.

The model was validated by two separate methods. First, the model's outputs were used to train and test a convolutional neural network to assess how well the model's output represents the input signal. Second, the scoring probabilities integrated from the model's output were used to create ROC curves and to calculate AUC scores. The purpose was to compare the performance of the validation network and the models' outputs to the base scoring probability models' AUC scores to find if the model improves the detection ability. Table 7.1 presents the AUC scores of the scoring probability models and OPSO models, as well as the AUC of the validation CNN. While the CNN couldn't reach the same AUC, the OPSO models' outputs did increase the AUC in comparison to the base scoring probability models. The model is thus considered a valid method to detect the off-puck scoring

Model	AUC
Logistic regression	0.81
Exponential distribution	0.81
Multilayer perceptron	0.83
OPSO (Logistic regression)	0.84
OPSO (Exponential distribution)	0.84
OPSO (Multilayer perceptron)	0.84
Validation CNN	0.72

Table 7.1. *AUC scores for the OPSO models, the base scoring probability models, and the validation CNN for predicting goal-scoring within 5 seconds.*

opportunities in ice hockey.

In conclusion, a new method for detecting off-puck scoring opportunities in ice hockey was developed. The objective was to create a method to detect and locate off-puck scoring opportunities based on position data. The objective was met with the OPSO model.

7.2 Future considerations

For future considerations, the model has room for improvement. The pitch control method used in this thesis does not take into consideration the direction the players are facing when they are moving. It would be intuitive that the influence a player projects onto the playing field is affected by the player's orientation as a player moving backward can influence the area they are moving away from more than a player facing forward. Another factor that could be considered is the handedness of players and how that influences their control area and the goal-scoring threat they produce depending on their positioning in contrast to the puck. Being able to take a shot directly from a pass should result in a higher goal-scoring probability than having to control the puck first let alone taking a backhanded one-timer.

Another improvement one should consider moving forward with OPSO is the scoring probability model. The models used in this thesis were based purely on the events' locations. More features could be represented by these models about the state of the game. For instance, the scoring probability increases if the effort on goal follows a lateral pass across the goal which could be represented as an increased probability to score on the opposite side of the rink. The passing probability component could also be improved by taking into account the availability of passing lanes toward the regions where teammates are headed. The current implementation does not consider possible intercepting players as it's based solely on the distance of the pass and the pitch control on the receiver location.

REFERENCES

[1] Smith, R. How Arsenal and Arsène Wenger Bought Into Analytics. eng. *New York Times (Online)* (2017). ISSN: 1553-8095. (Visited on 11/21/2023).

[2] Spearman, W. Beyond Expected Goals. (Mar. 2018). URL: https://www.researchgate.net/publication/327139841_Beyond_Expected_Goals (visited on 02/21/2024).

[3] Fernandez, J. and Bornn, L. Wide Open Spaces: A statistical technique for measuring space creation in professional soccer. (2018). URL: https://www.researchgate.net/publication/324942294_Wide_Open_Spaces_A_statistical_technique_for_measuring_space_creation_in_professional_soccer (visited on 02/21/2024).

[4] Gavião, L. O., Sant'Anna, A. P., Alves Lima, G. B. and Almada Garcia, P. A. de. Evaluation of soccer players under the Moneyball concept. eng. *Journal of sports sciences* 38.11-12 (2020), pp. 1221–1247. ISSN: 0264-0414.

[5] Macdonald, B. An Expected Goals Model for Evaluating NHL Teams and Players. *Proceedings of the 2012 MIT Sloan Sports Analytics Conference* (Jan. 2012). URL: https://www.researchgate.net/publication/236687040_An_Expected_Goals_Model_for_Evaluating_NHL_Teams_and_Players (visited on 01/23/2024).

[6] Ryder, A. *Shot Quality, a methodology for the study of the quality of a hockey team's shots allowed.* Tech. rep. Online article. 2004. URL: http://hockeyanalytics.com/Research_files/Shot_Quality.pdf (visited on 01/24/2024).

[7] Wu, L. and Swart, T. B. *A New Metric for Pitch Control based on an Intuitive Motion Model*. Tech. rep. Online article. May 2023. URL: https://www.sfu.ca/~tswartz/papers/pitch_control.pdf (visited on 11/13/2023).

[8] Zafari, F., Gkelias, A. and Leung, K. K. A Survey of Indoor Localization Systems and Technologies. eng. *IEEE Communications surveys and tutorials* 21.3 (2019), pp. 2568–2599. ISSN: 1553-877X.

[9] Gyula, S. *Recent Advances in Indoor Localization Systems and Technologies*. eng. Basel, Switzerland: MDPI - Multidisciplinary Digital Publishing Institute, 2021. ISBN: 3-0365-1483-X.

[10] Park, J., Kim, H., Yoon, J., Kim, H., Park, C. and Hong, D. Development of an ultrasound technology-based indoor-location monitoring service system for worker safety in shipbuilding and offshore industry. eng. *Processes* 9.2 (2021), pp. 1–16. ISSN: 2227-9717.

[11] Ebner, F. *Smartphone-based 3D indoor localization and navigation*. eng. Human Data Understanding - Sensors, Models, Knowledge. Berlin: Logos Verlag Berlin, 2021. ISBN: 3-8325-8623-7.

[12] Gupta, N. *Inside Bluetooth low energy*. eng. Second edition. Artech House mobile communciations series. London: Artech House, 2016. ISBN: 1-5231-3476-3.

[13] Walpole, R. E., Myers, R. H., Myers, S. L. and Ye, K. *Probability & statistics for engineers & scientists*. eng. Ninth edition. Boston: Pearson, 2016. ISBN: 978-1-292-16141-9.

[14] Scikit-learn. *Logistic function*. URL: `https://scikit-learn.org/stable/auto_examples/linear_model/plot_logistic.html` (visited on 10/23/2023).

[15] Hastie, T., Tibshirani, R. and Friedman, J. *Elements of Statistical Learning: Data Mining, Inference, and Prediction*. eng. Second Edition. Springer Series in Statistics. New York: Springer, 2009. ISBN: 0387848576.

[16] Sewak, M., Karim, R. and Pujari, P. *Practical Convolutional Neural Networks*. eng. 1st edition. Packt Publishing, 2018. ISBN: 1-78839-230-2.

[17] Vang-Mata, R. *Multilayer perceptrons : theory and applications*. eng. Computer science, technology and applications. New York: Nova Science Publishers, 2020. ISBN: 1-5361-7365-7.

[18] Lee, J. H., Delbruck, T. and Pfeiffer, M. Training deep spiking neural networks using backpropagation. eng. *Frontiers in neuroscience* 10 (2016), pp. 508–508. ISSN: 1662-4548.

[19] Michelucci, U. *Advanced Applied Deep Learning Convolutional Neural Networks and Object Detection*. eng. 1st ed. 2019. Berkeley, CA: Apress, 2019. ISBN: 1-4842-4976-3.

[20] Hanley, J. and McNeil, B. The meaning and use of the area under a receiver operating characteristic (ROC) curve. eng. *Radiology* 143.1 (1982), pp. 29–36. ISSN: 0033-8419.