NBA Oracle

Anderson Compalas
Cindy Wong
Caleb Cho

Concept

Pytorch model to predict winners in NBA games

Using both teams' rolling averages and who is playing home or away

How does it relate to the theme?

Sports Betting

Types of Bets

Crypto Sports Betting

Straight Bets

Point Spreads

Total Line Bets

Moneyline

Teasers

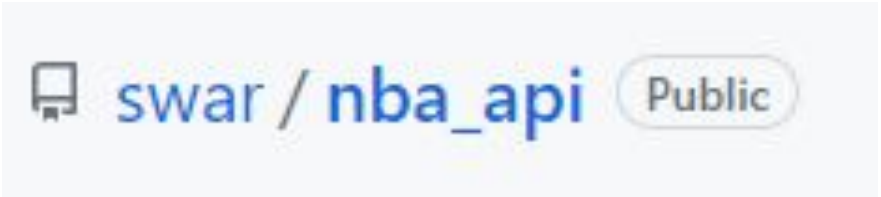
Prop bets

Head-to-head bets

Parlays

Resources

- Website: Basketball Reference
- What: Comprehensive data of 17,000 different games from 2015 - 2022
- Why: Train and test model



- API: nba_api
- What: Live data for 2023
- Why: Make predictions on upcoming matches

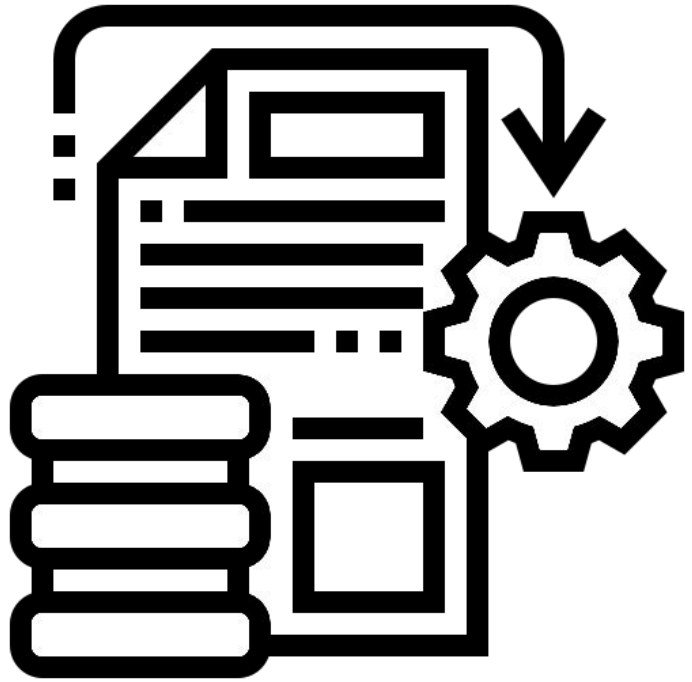
Preprocessing

Basketball Reference

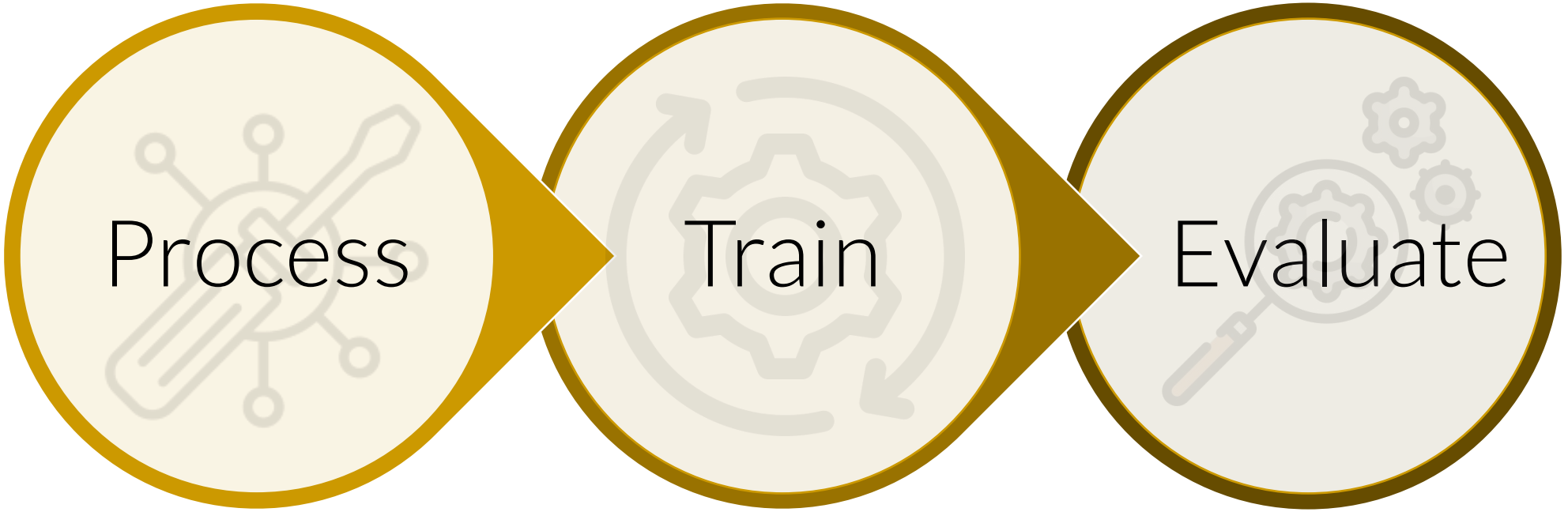
- Sorted games by date
- Removed irrelevant columns
- Removed columns with null values
- Added target column
- Calculated rolling averages

NBA_API

- Removed irrelevant columns
- Formatted team names with abbreviations
- Added columns for season, date, and time
- Concatenated most recent rolling averages for each team



Model



Training Process

Feature Selection:

Extracted relevant numerical columns or features from the processed data frame as training features

Data splitting:

Separated data by season into training and testing sets

Data Transformation:

Convert selected data into pytorch tensors

Training Loop

```
Epoch 1/20, Loss: 0.6431, Accuracy: 62.73%,
Epoch 2/20, Loss: 0.6302, Accuracy: 64.09%,
Epoch 3/20, Loss: 0.6243, Accuracy: 64.90%,
Epoch 4/20, Loss: 0.6212, Accuracy: 65.79%,
Epoch 5/20, Loss: 0.6176, Accuracy: 65.73%,
Epoch 6/20, Loss: 0.6123, Accuracy: 66.55%,
Epoch 7/20, Loss: 0.6081, Accuracy: 66.67%,
Epoch 8/20, Loss: 0.6034, Accuracy: 67.19%,
Epoch 9/20, Loss: 0.5981, Accuracy: 67.77%,
Epoch 10/20, Loss: 0.5966, Accuracy: 67.58%,
Epoch 11/20, Loss: 0.5909, Accuracy: 68.37%,
Epoch 12/20, Loss: 0.5862, Accuracy: 69.13%,
Epoch 13/20, Loss: 0.5839, Accuracy: 68.57%,
Epoch 14/20, Loss: 0.5782, Accuracy: 69.67%,
Epoch 15/20, Loss: 0.5750, Accuracy: 69.81%,
Epoch 16/20, Loss: 0.5725, Accuracy: 69.83%,
Epoch 17/20, Loss: 0.5690, Accuracy: 70.24%,
Epoch 18/20, Loss: 0.5609, Accuracy: 71.00%,
Epoch 19/20, Loss: 0.5566, Accuracy: 71.59%,
Epoch 20/20, Loss: 0.5548, Accuracy: 71.57%,
```

BCE LOSS

- Well suited for binary classification (0 or 1)
- Quantifies the difference between the predicted probabilities (after applying a sigmoid activation)

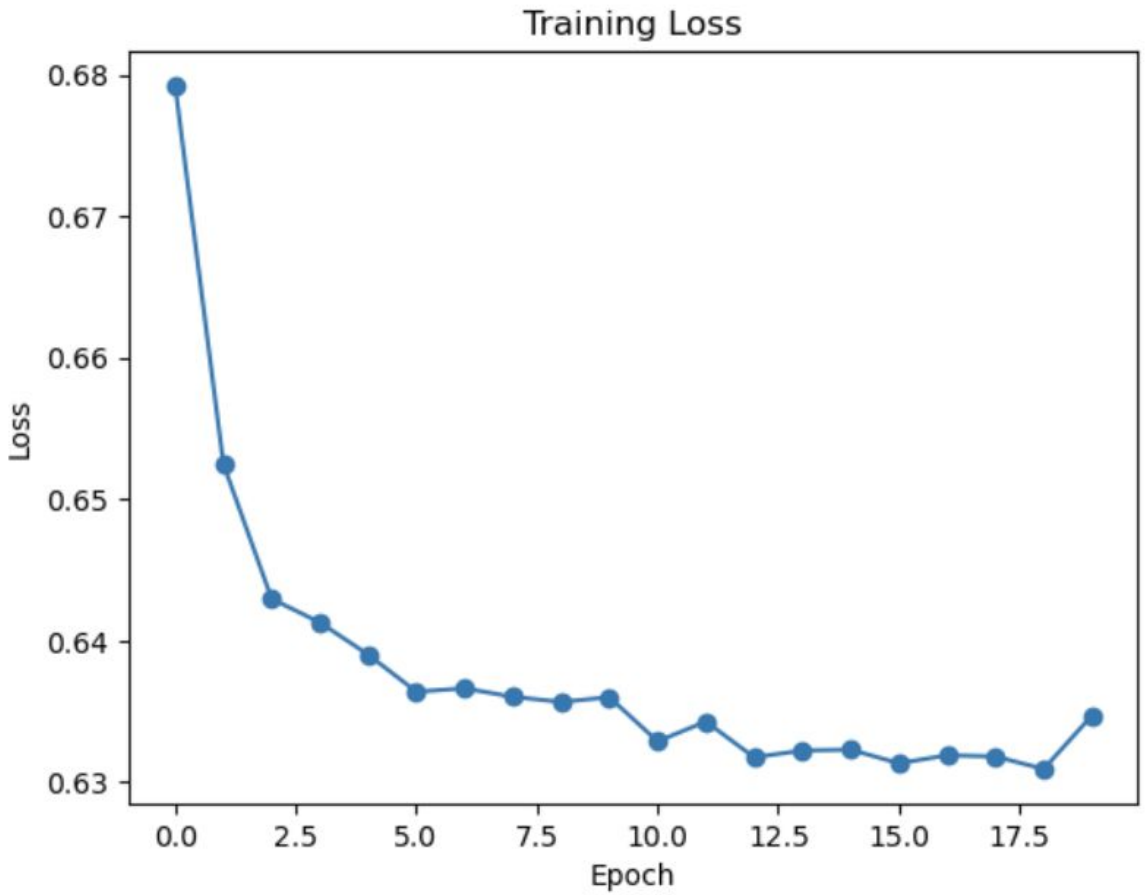
Optimizer

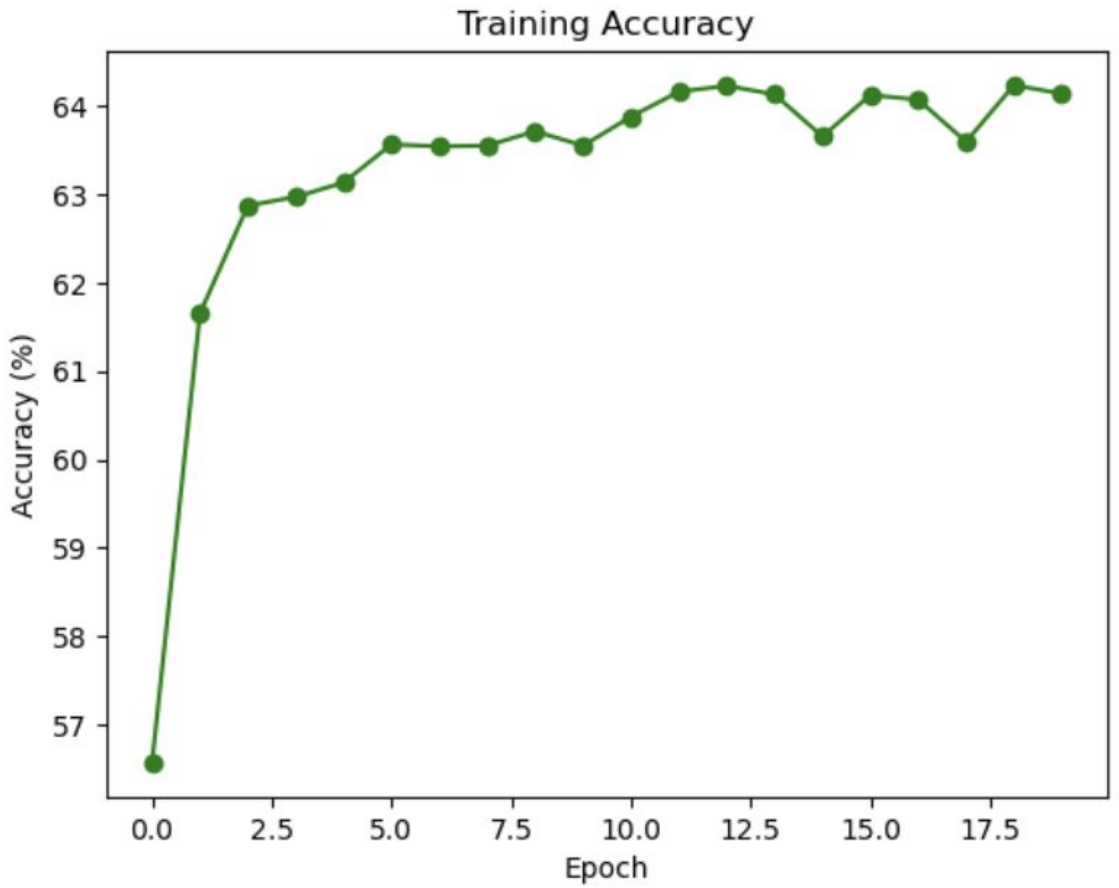
- Adam()
- The default optimizer for ML engineers is Adam() as it is quick to converge.

Hyperparameters

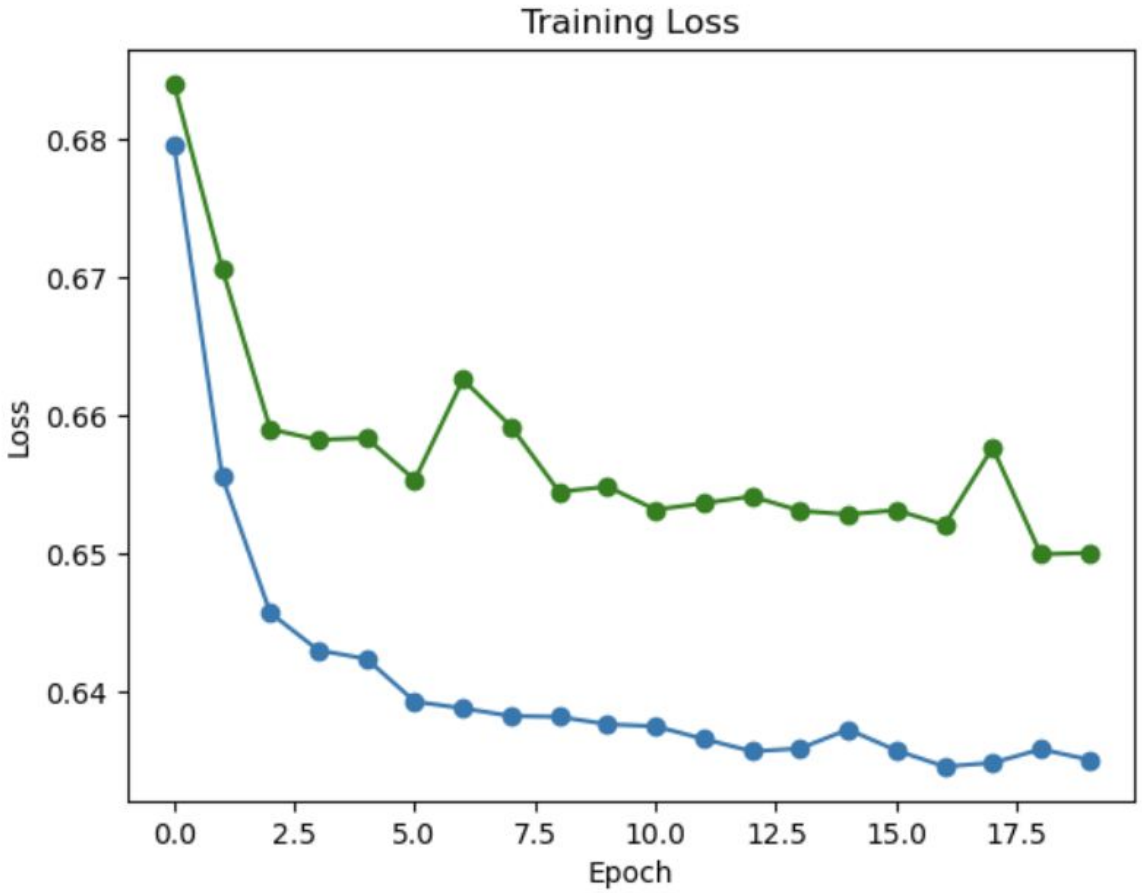
- Epochs: 20
- Lr: 0.001

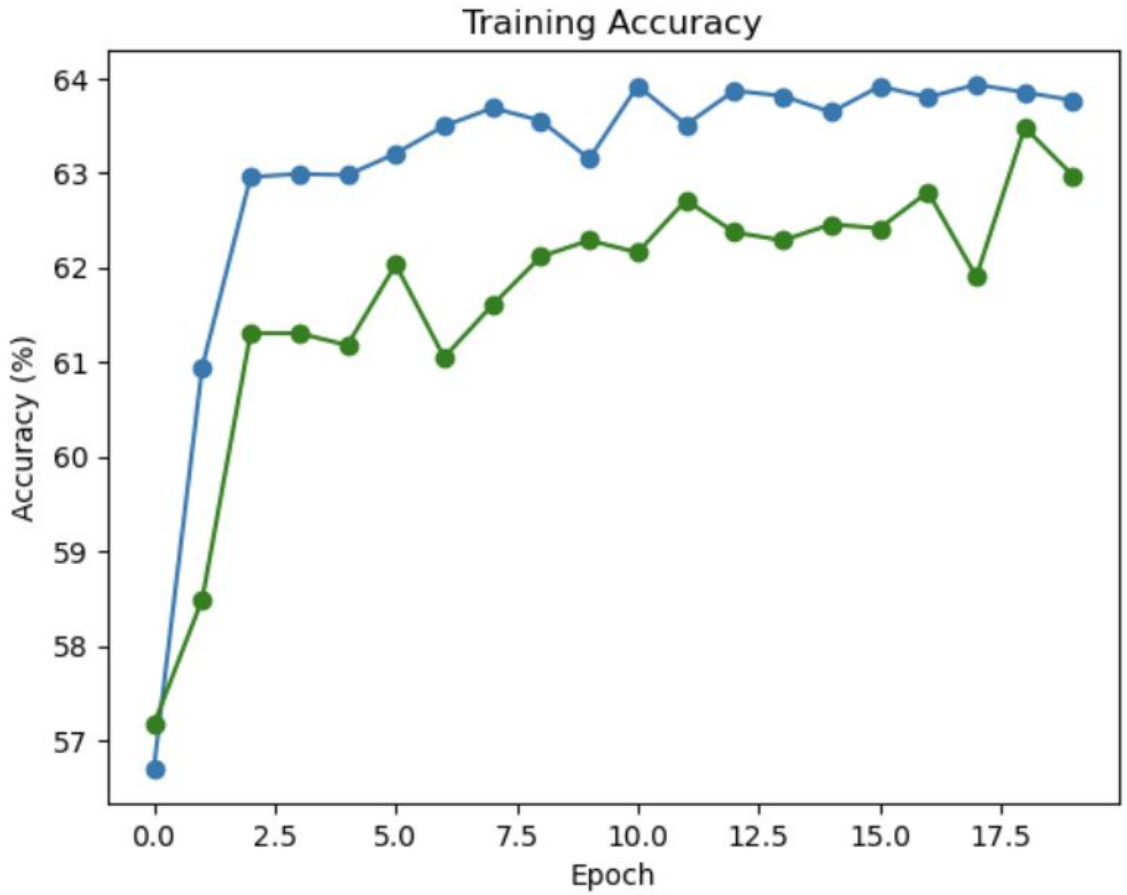
Training Loss and Accuracy





Testing and Validation



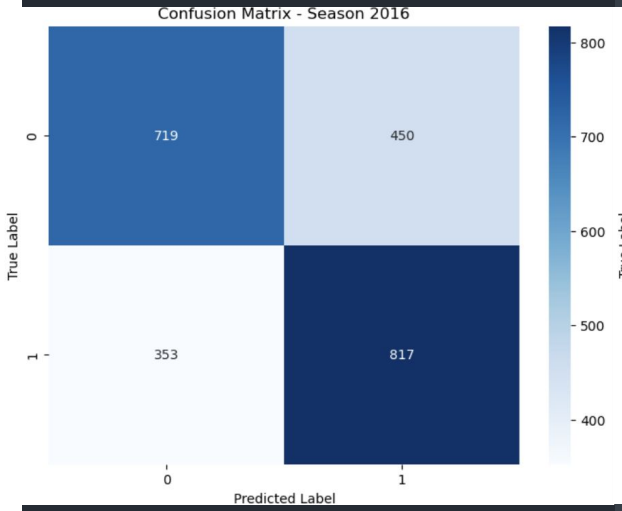


Testing

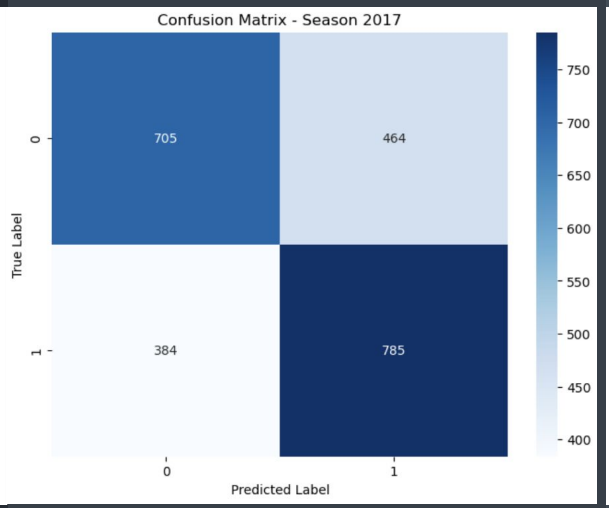
Backtest Function:

Checks the accuracy of each season and visualizes using confusion matrices

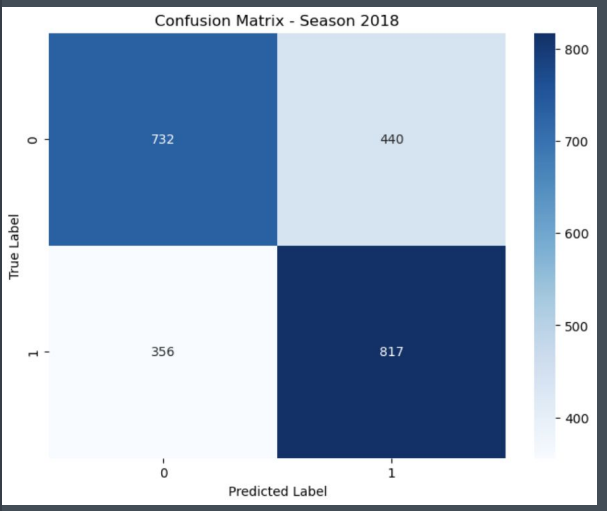
2016



2017

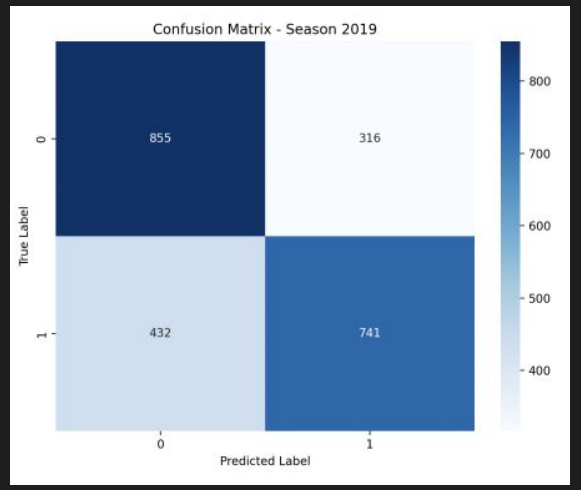


2018

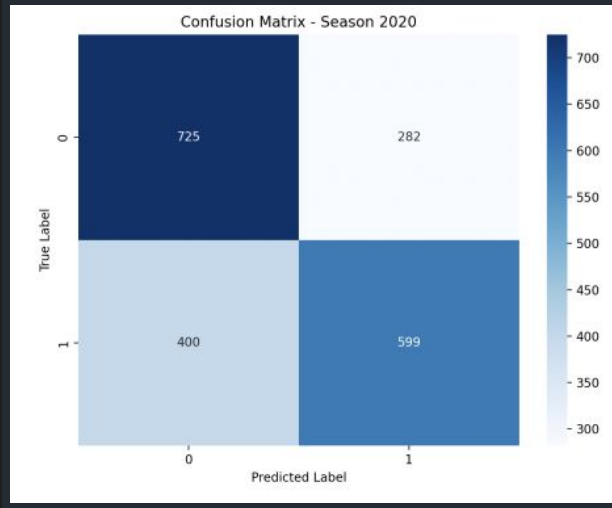


Testing

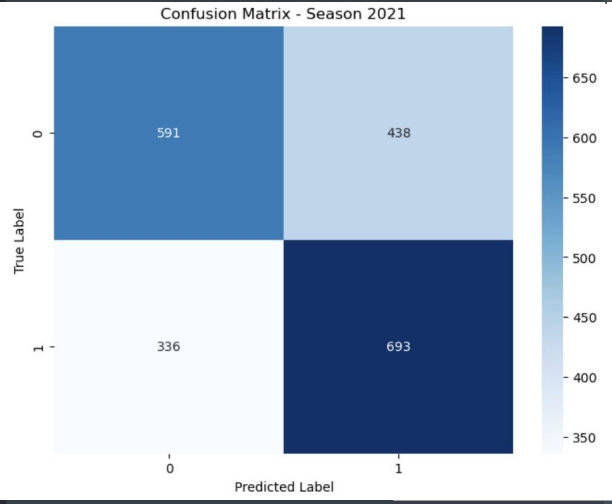
2019



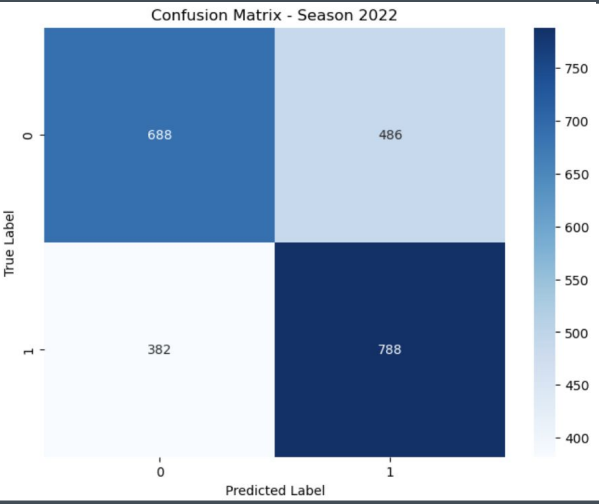
2020



2021



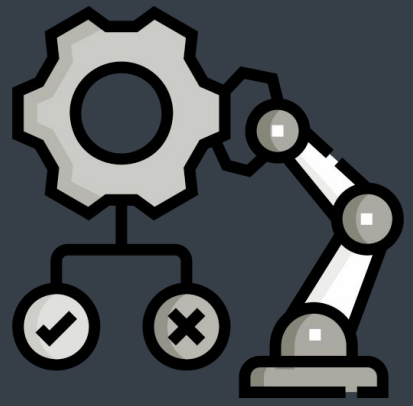
2022



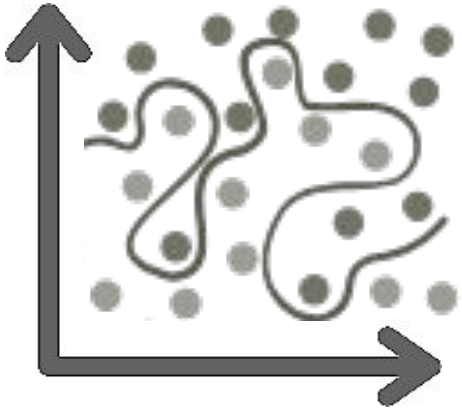
Challenges

Loss not decreasing while accuracy increasing

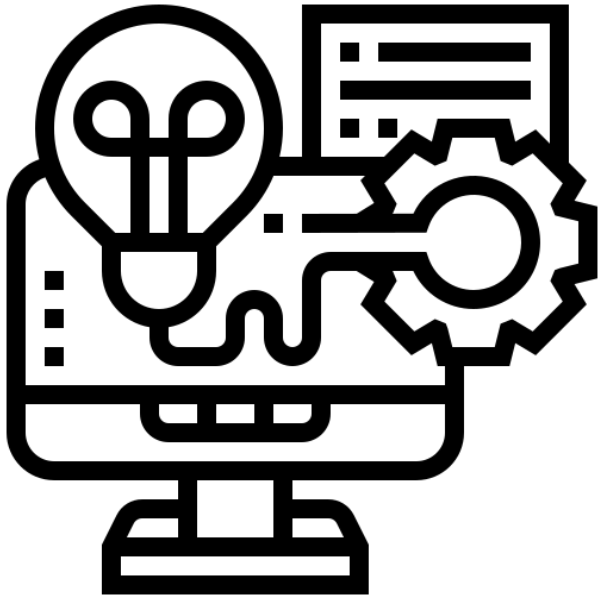
Great training accuracy but poor test accuracy



Both are signs of model overfitting



Overcoming Challenges



Hyperparameter tuning

- Learning rate scheduling:
- Updated learning rate based off of average loss.
- Epoch adjustment:
- Found that fewer epochs led to higher accuracy.

Batch Normalization Layer:

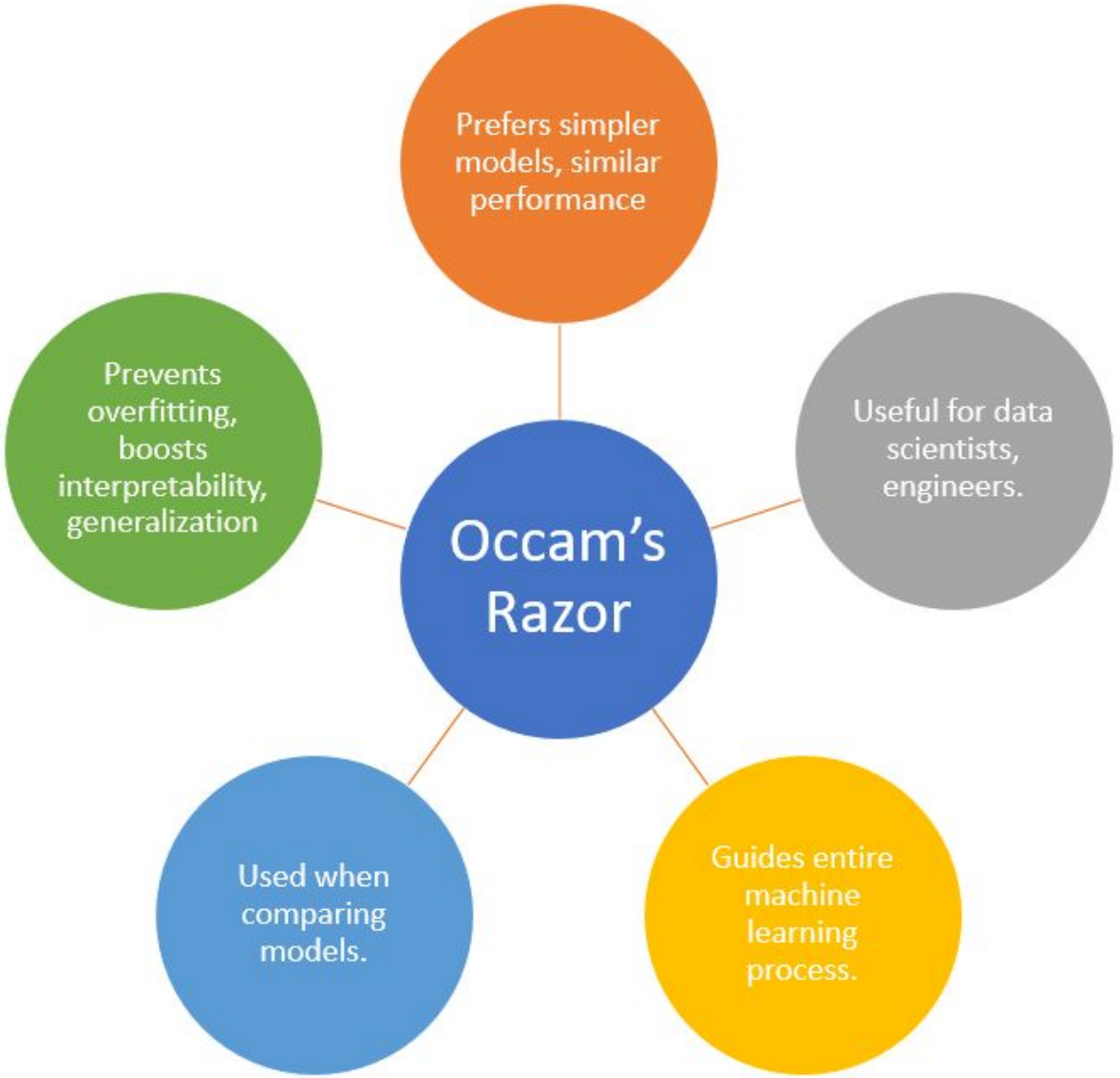
- Normalizes the input to each neuron (or activation) within a layer.
- Acts as a regularizer introducing slight noise during training.
- Stabilizes training and accelerates convergence.

Dropout Layer

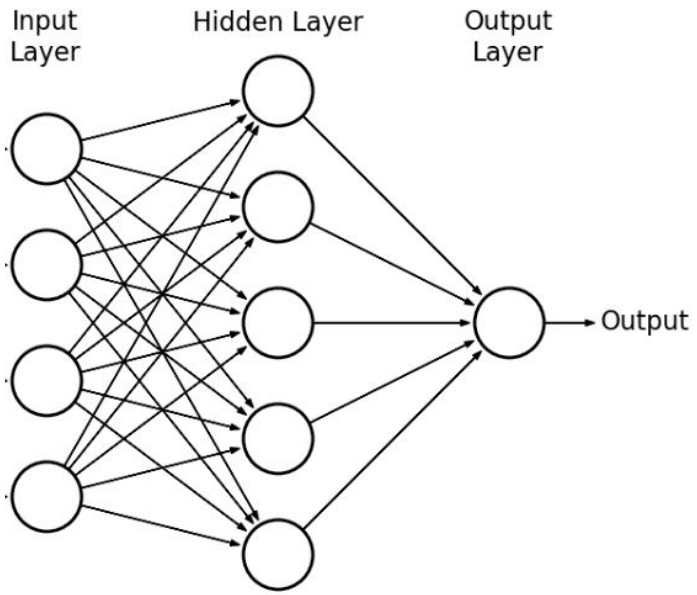
- Prevents overfitting by randomly setting a fraction of input units to zero during each training iteration.
- By introducing randomness, dropout encourages the network to be more robust and less prone to memorizing training data.

Improving Accuracy

- Simplifying model:
  - Applied Occam's Razor,
  - Simpler model led to higher test accuracy.
- Outperformed other attempted techniques to improve accuracy and prevent overfitting.



Architecture



Hidden

Fully Connected Layer (fc1):

Number of Neurons: 128

Function: Linear(input_size, 128)

This layer performs a linear transformation on the input features.

Hidden

Relu Activation function:

This layer introduces non-linearity by applying the Rectified Linear Unit (ReLu) activation element_wise to the output

Output

Fully Connected Layer 2:

self.fc2 = nn.Linear(128, 1)

Sigmoid Activation Function: Sigmoid()

Squashes the output to a range between 0 and 1 to make it appropriate for binary classification.

NBA Oracle

Home	Away		Home Win %	Away Lose %
Select Team	Select Team	Submit	----	----

Home	Away	Date	Winner
POR	UTA	2023-12-03	POR
SAC	DEN	2023-12-03	SAC
LAL	HOU	2023-12-03	LAL
GSW	LAC	2023-12-02	GSW
CHA	MIN	2023-12-02	CHA
BKN	ORL	2023-12-03	BKN
DET	CLE	2023-12-03	DET
MIA	IND	2023-12-03	MIA
CHI	NOP	2023-12-03	CHI

https://fluoridated-tin-hydrofoil.glitch.me/

NBA Oracle

Home | Away | | Home Win % | Away Lose %

Select Team | Select Team | Submit | ---- | ----

Select Team
POR
SAC
LAL
GSW
CHA
BKN
DET
MIA
CHI
MIL
DAL

	Away	Date	Winner
	UTA	2023-12-03	POR
	DEN	2023-12-03	SAC
	HOU	2023-12-03	LAL
	LAC	2023-12-02	GSW
	MIN	2023-12-02	CHA
	ORL	2023-12-03	BKN
DET	CLE	2023-12-03	DET
MIA	IND	2023-12-03	MIA
CHI	NOP	2023-12-03	CHI

https://fluoridated-tin-hydrofoil.glitch.me/

NBA Oracle

Home	Away		Home Win %	Away Lose %
SAC	Select Team	Submit	----	----

Select Team
UTA
DEN
HOU
LAC
MIN
ORL
CLE
IND
NOP
ATL
OKC

Home		Date	Winner
POR		2023-12-03	POR
SAC		2023-12-03	SAC
LAL		2023-12-03	LAL
GSW		2023-12-02	GSW
CHA		2023-12-02	CHA
BKN		2023-12-03	BKN
DET	CLE	2023-12-03	DET
MIA	IND	2023-12-03	MIA
CHI	NOP	2023-12-03	CHI

https://fluoridated-tin-hydrofoil.glitch.me/

NBA Oracle

Home	Away		Home Win %	Away Lose %
SAC	DEN	Submit	----	----

Home	Away	Date	Winner
POR	UTA	2023-12-03	POR
SAC	DEN	2023-12-03	SAC
LAL	HOU	2023-12-03	LAL
GSW	LAC	2023-12-02	GSW
CHA	MIN	2023-12-02	CHA
BKN	ORL	2023-12-03	BKN
DET	CLE	2023-12-03	DET
MIA	IND	2023-12-03	MIA
CHI	NOP	2023-12-03	CHI

https://fluoridated-tin-hydrofoil.glitch.me/

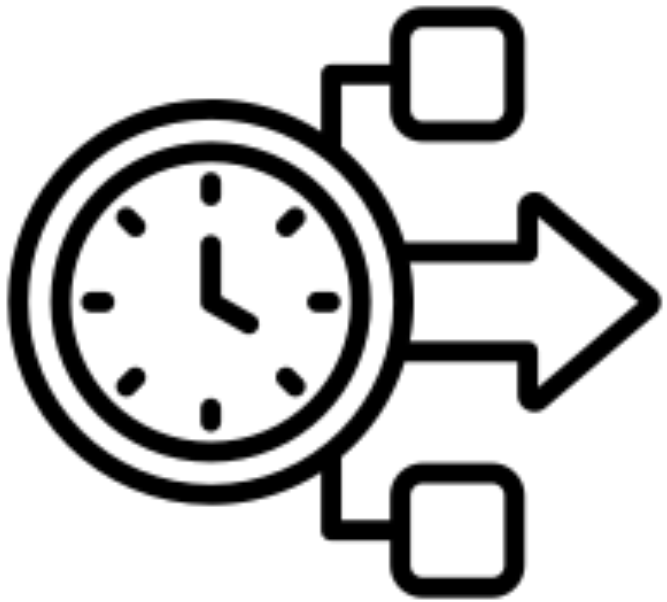
NBA Oracle

Home	Away		Home Win %	Away Lose %
SAC	DEN	Submit	51.07%	48.93%

Home	Away	Date	Winner
POR	UTA	2023-12-03	POR
SAC	DEN	2023-12-03	SAC
LAL	HOU	2023-12-03	LAL
GSW	LAC	2023-12-02	GSW
CHA	MIN	2023-12-02	CHA
BKN	ORL	2023-12-03	BKN
DET	CLE	2023-12-03	DET
MIA	IND	2023-12-03	MIA
CHI	NOP	2023-12-03	CHI

https://fluoridated-tin-hydrofoil.glitch.me/

Future Plans



Connect website with backend instead of manually entering in data after running code

Fix website code so that it doesn't become warped on different screens

Include training data up to 2023.

Thank you for listening

Public repo: https://github.com/acompalas/SCAI_Hedge_your_bets