ENPM808A

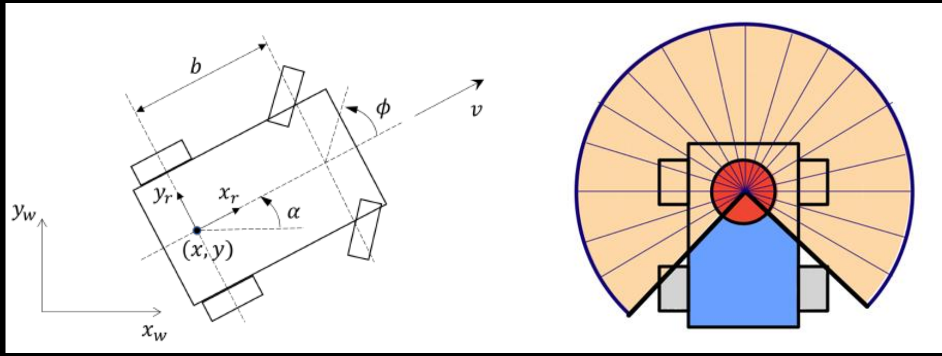
MACHINE LEARNING FINAL PROJECT

By

Jai Sharma

THE PROBLEM

We are provided with a novel dataset on which we need to perform regression. We are given Lidar, Pose and Goal information. We have to use a regression model to predict the command velocity (translational and angular) based on the path taken.



Features:

- *Laser Range*
- *Final Goal (x,y,qk,qr)*
- *Local Goal (x,y,qk,qr)*
- *Pose (x,y,qk,qr)*

Predict:

- *Cmd_vel_v*
- *Cmd_vel_w*

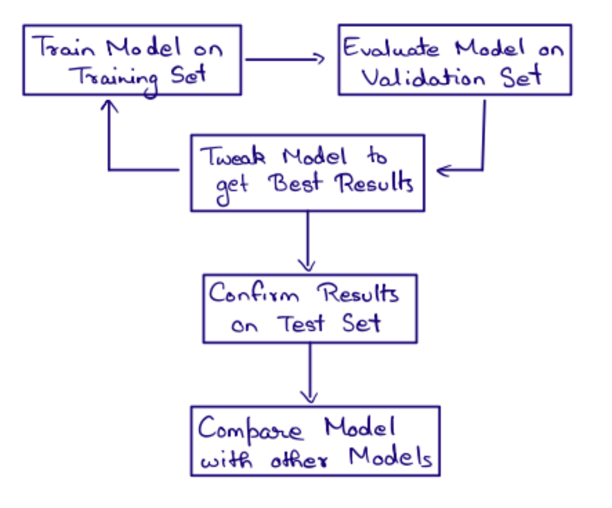
Data Processing

- Imputation Check
- Outlier Check
- Train-Validation Split

Feature Engineering

- Feature Selection
- Feature Reduction
- Feature Scaling

Machine Learning Pipeline



- The strategy was as follows:
- Train model with default parameters
- Run prediction on Training Data and Testing Data
- Compute R^2 score and Mean Square Error
- Tune Hyperparameters
  - Build parameter dictionary
  - Fit data on Validation Set to find best parameters
- Evaluate best model on Training Data and Testing Data
- Compute R^2 score and Mean Square Error for best version

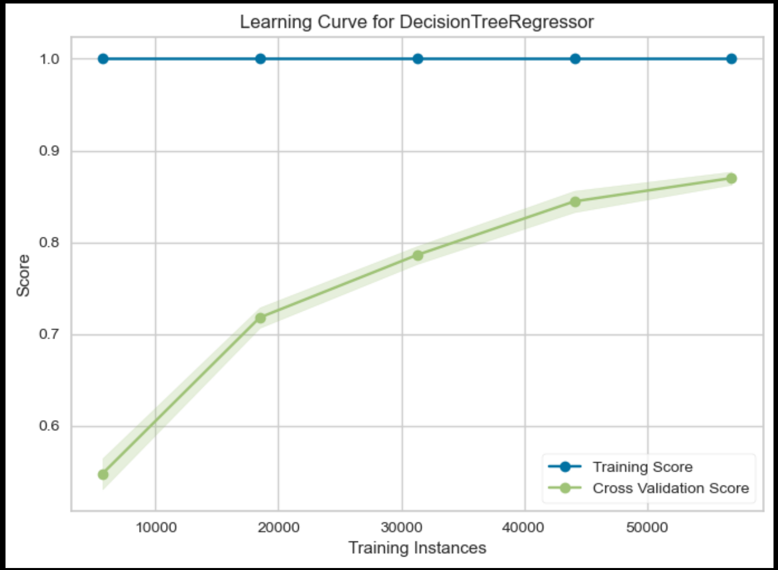
EVALUATION METRIC

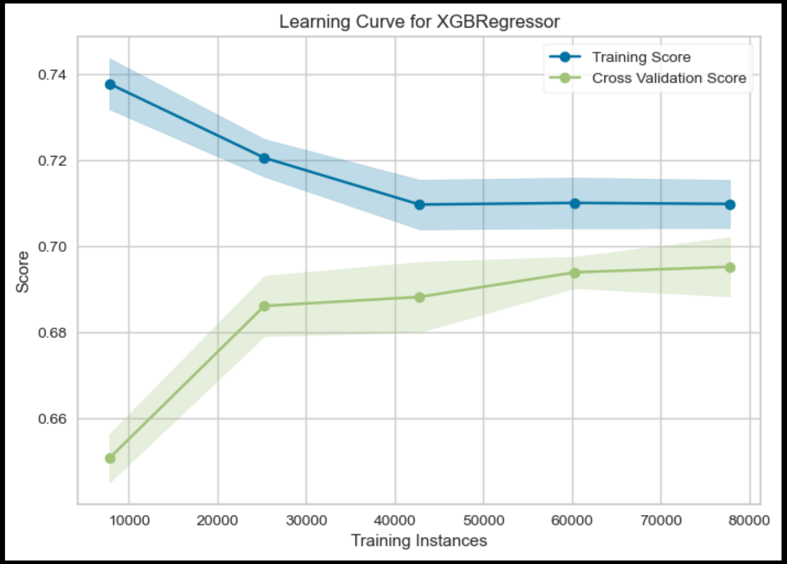
$$R^2 = 1 - \frac{SS_{Regression}}{SS_{Total}}$$

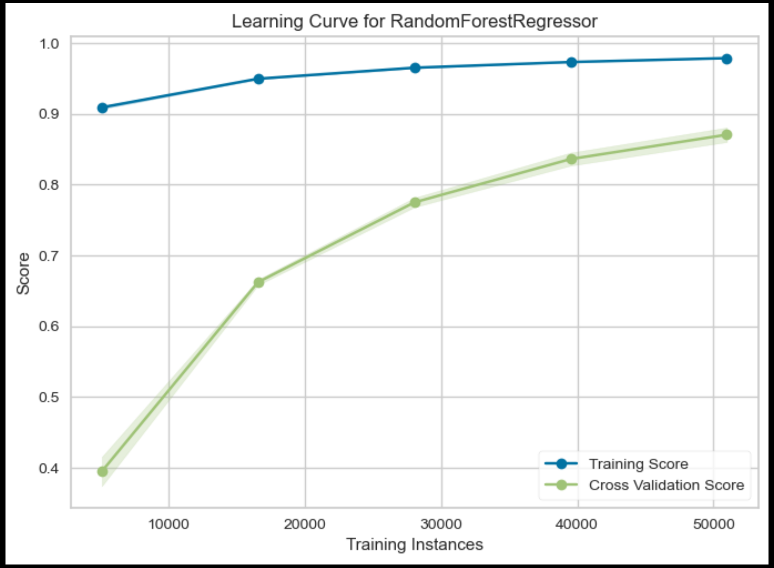
$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \tilde{y}_i)^2$$

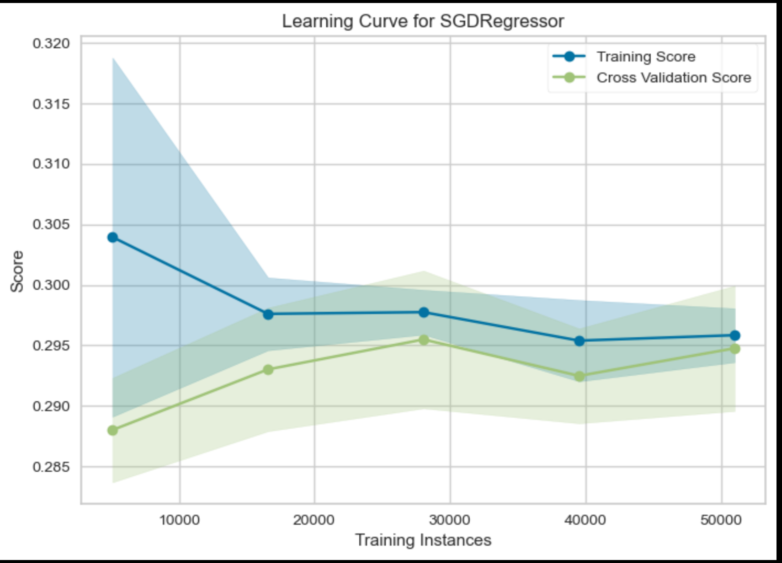
1. R-Squared
2. Mean Squared Error

Learning Curves









Conclusion: A training data set exceeding 50,000 instances is sufficient for this Learning Problem.

Regularization Strategies

Linear Regression with SGD:

a. penalty: Lasso Regularization (L1) and Ridge Regularization (L2) are considered.
b. alpha: tunes the regularization term. High value means high regularization.

Decision Tree, Random Forest and XGBoost:

a. max_Depth: reduce the maximum depth of the tree to discourage memorization
b. n_estimators: reduce number of decision trees to improve generalization
c. max_depth: specify a value to ensure that $\boldsymbol{E_{in}}$ is not 100 %

Models Considered

- Linear Regression with Stochastic Gradient Descent
- Decision Tree
- Random Forest
- XG Boost

CONCLUSIONS

- Tree based regression outperforms classic linear regressors.
- Tree based regression requires extensive hyperparameter tuning, they tend to overfit.
- Translational Velocity is more predictable and learnable given the features provided in the dataset.
- Extremely large datasets (> 100,000 instances) have no significant improvement on E_{out} value for any of the models.
- Randomized Search is faster than Grid Search but it may not always give the ideal hyperparameter configuration.

Best Model ==> XG Boost

XG Boost	Translational Velocity (v)		Angular Velocity (w)	
	Test Set 1	Test set 2	Test Set 1	Test set 2
MSE	0.01	0.09	0.02	0.10
R^2	0.92	0.34	0.77	-0.73

- XG Boost gives the best R^2 scores while giving the lowest MSE values.
- The Out of Sample errors are shown on the right:
- Other models are indeed easier to understand and implement.
- However, the immense amount of data, and the complexity of the learning problem required a learning architecture that is efficient and robust.
- Hence, XG Boost came out as the winner.

THANK YOU