

# MS5

Leo Qian, Ronald Feng, Aydan Gerber, Sucheer Maddury

## Questions

- **What, if any, insights did you get from your PM meeting?**

We were unable to meet with our PM this week but have scheduled a meeting with him for next week. We plan to show him our progress on supervised learning and ask for his recommendations and feedback about what is a good next step for us to take for our project.

- **What challenges arose this week?**

Many of us were busy this week with prelims so it was difficult to schedule the work, but we still were able to complete the milestone.

- **What went well?**

Based on our data, we were able to tackle the regression problem easily as it fit the question we already had.

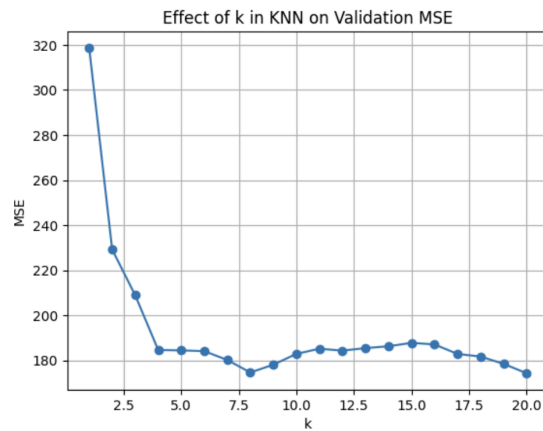
- **What do you want to do for the last week of your project? (April 12 - April 19)**

We are not sure but we are in the middle of discussing this. As of now, we are thinking about improving the regression models we have.

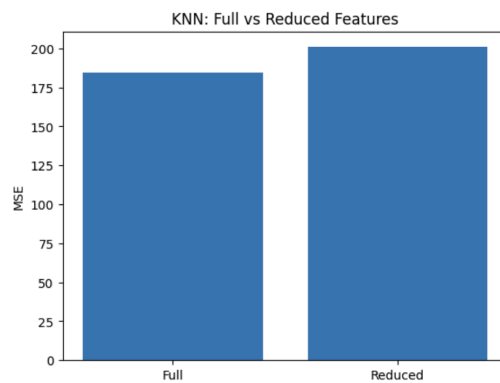
## Supervised Learning

We chose to tackle the “Supervised Regression” problem as that fits the data that we have.

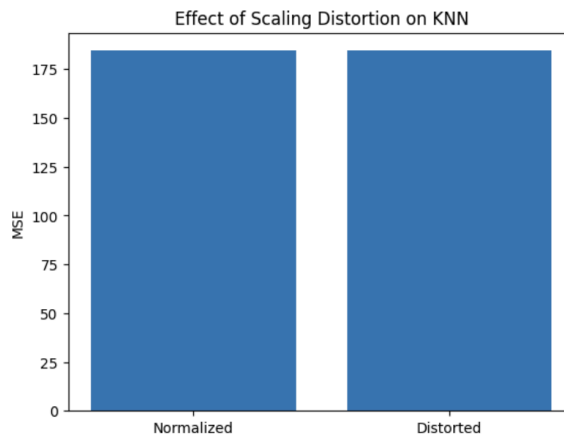
- a. See #KNNR
  - i. KNN finds the k closests data points by distance (neighbors) and averages the target values of those neighbors to predict the target values for the validation set.
  - ii. Making K larger decreases the MSE, due to better generalization, however smaller K allows for more specificity but is prone to overfitting.



- iii. Reducing features hurts performance overall, likely because of less data, so the model is too generalized, which hurts accuracy.

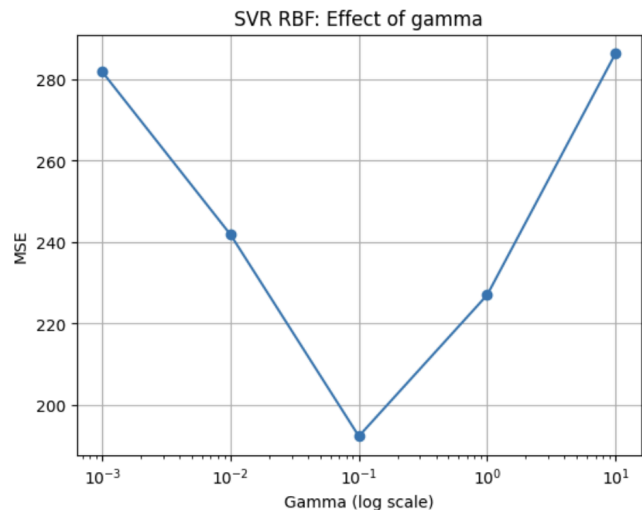
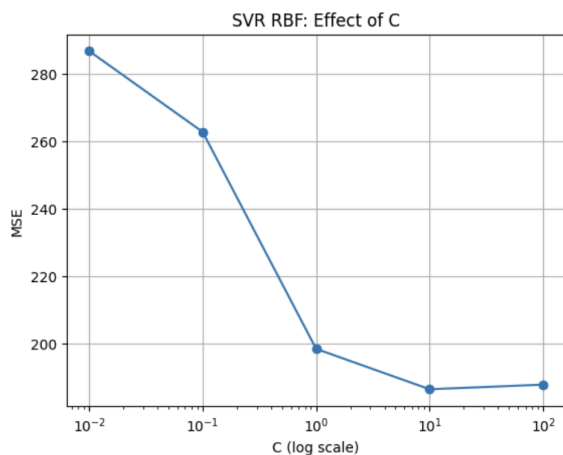


- iv. No change, since KNN depends on distance, the scaling does not change the relative distance between data points.



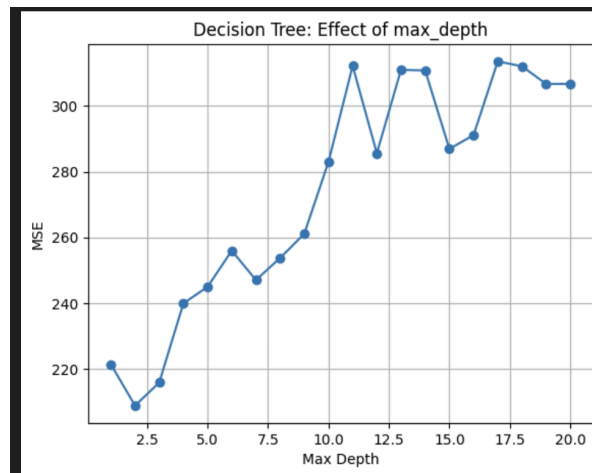
b. See #SVR

- i. SVR fits a close-to-linear function that stays within margins of truth values of training data.
- ii. It is similar to LR in that both try to make predictions for problems with linear output. However, it can use kernels for higher-dimensional data.
  - 1. Linear kernel
    - a. For linearly separable data,
    - b. Behaves like LR
  - 2. Polynomial kernel
    - a. Higher dimensional polynomial space
    - b. Curves for complex data
  - 3. Rbf Radial Basis Function
    - a. Infinite dimensional spaces
    - b. Non-linear relationships
- iii. A larger C results in a tighter margin, as there is a larger penalty for errors, which can result in overfitting. This is seen across all Kernels, but RBF is displayed below. Larger gamma values do not seem to improve performance, there seems to be a sweet spot where the margins are just right and error is minimized (on our data), which can be found with hyperparameter tuning. The increase in MSE as gamma increases is likely due to overfitting, as gamma determines how tight the model fits around data.

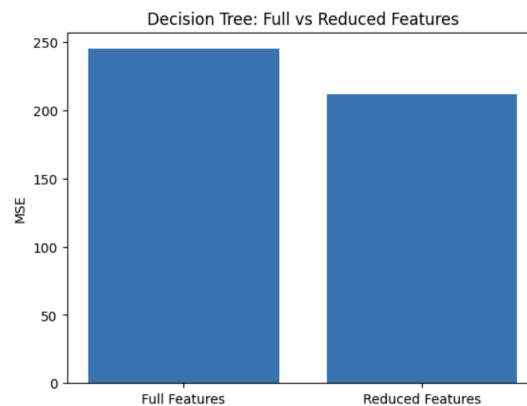


c. SEE #DTR

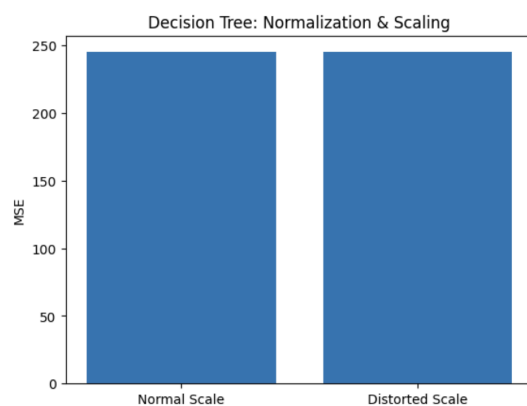
- i. It splits data into regions to predict the continuous value recursively and reducing MSE. Each node makes a decision based on a feature value range and each leaf contains average target value for the data in the group.
- ii. With a low max depth, the model underfits, with a high max depth, the model overfits.



- iii. With less features, there can be less noise, which may help accuracy by improving generalization. However if the feature is important then accuracy can be lost.

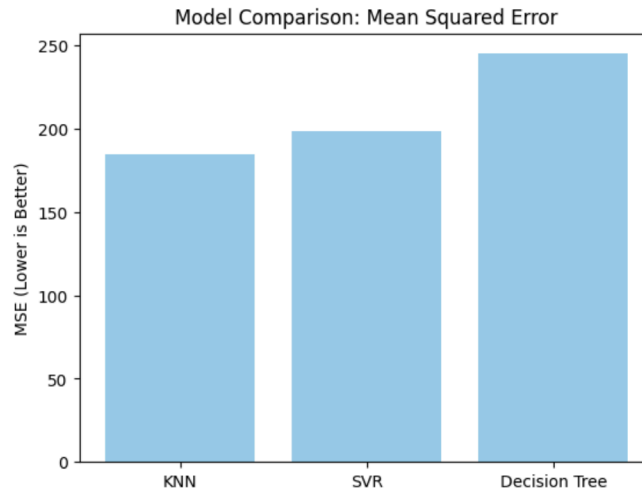


- iv. Don't need normalization, so distortion has no effect. This is because the trees split based on thresholds instead of specific values or distances.



d. Additional questions

- i. Out of the models, KNN performed the best, with lowest MSE and KNN was able to identify localized patterns within clusters of wins / losses in the data. Whereas the other two must find larger trends.



ii. Some trade offs:

1. KNN doesn't require training, so it will be fastest
2. Decision tree is fastest, as the tree is built and requires few resources to follow
3. Generally depends on data, but probably KNN since it works with localized points
4. Generally depends on data, but probably also KNN since it works with localized points and can also work with larger clusters to minimize loss. Since SVR has many hyperparameters it's also possible it can be made to be better at validation (generalization)

## Signatures

We have all read this report and assent to its contents.

Leo Qian  
Ronald Feng  
Aydan Gerber  
Sucheer Maddury