Model Evaluation
+
Bias-Variance Trade-off

Pantelis Tzamalis

Ph.D. Candidate, University of Patras, Greece, Computer Engineering and Informatics Department
Research Engineer, Computer Technology Institute & Press “Diophantus”, Gree©ce

Topics presented in these slides

Model Evaluation is a fundamental topic of understanding your model's performance.

The following topics will be presented:

- Train Test Split
- Holdout Sets
- Parameter Grids
- Scala and Spark for Model Evaluation
- Bias Variance Trade-Off

Model Evaluation

- When you train a Machine Learning Algorithm on some data, afterwards you will want some measure of how well it performed.
- Each main Machine Learning Task has different metrics for evaluation:
  - Regression
    - R^2
    - RMSE
  - Classification
    - Precision
    - Recall
  - Clustering
    - Within Sum of Squares Error

Train/Test Split

1. It is not a good idea to get these measurements using the same data you trained your model on.
2. Your model has already seen this data, which means it is not a good choice for evaluating your model's performance
3. **You should get these metrics on test data, which your model has not seen yet.**
4. This is known as a **train-test split**.

Holdout Data

- An expansion of this idea is the **holdout data set**.
- This is separate from the training and test sets.
- *In this process, you use the training data to fit your model, you use the test set to evaluate and adjust your model.*
- You can use the test set over and over again.
- *Finally, before deploying your model, you check it against the holdout to get some final metrics on performance.*

Parameter Grids

- You can add optional parameters to Machine Learning Algorithms.
- Many times it is difficult to know which could be the good values for these parameters.
- **Spark** makes it possible to set up a grid of parameters to train across.
- You create multiple models, then you train them across the grid, and Spark reports back which model performed the best results.

Later on, in the next tutorials, I'll present you some other methods to evaluate the best parameters values on a model, without using Spark.

Spark for Model Evaluation

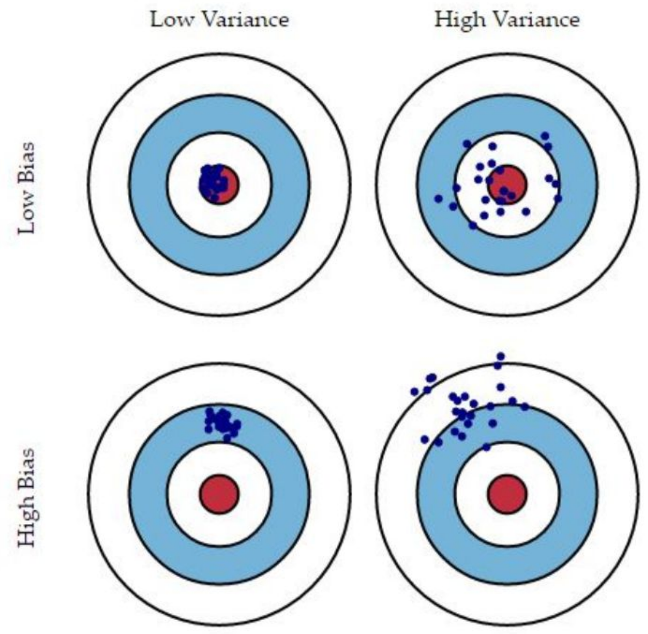
- Spark makes all of these processes generally easy with the use of 3 object types:
    - Evaluators
    - ParamGridBuilders
    - TrainValidationSplit

An important aspect of understanding all of this is the Bias-Variance Trade-Off, which will be presented in the next slides. Exploring this with Spark pertains more to theory than Data Engineering, however, the next slides explain the concept so you can have a full context for this topic.

Bias Variance Trade-Off

- The bias-variance trade-off is the point where you are adding just noise by adding model complexity (flexibility).
- The training error goes down as it has to, but the test error is starting to go up.
- The model after the bias trade-off begins to overfit.

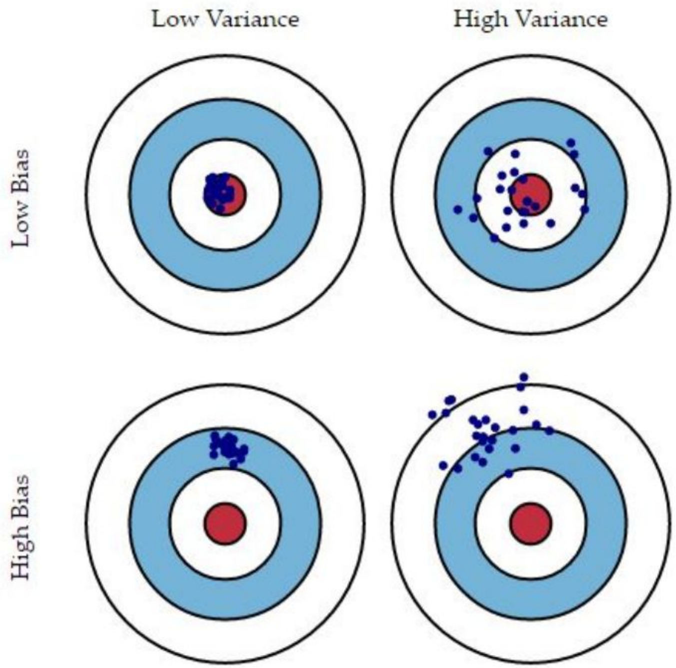
- Imagine that the center of the target is a model that perfectly predicts the correct values.
- As we move away from the bulls-eye, our predictions get worse and worse.



Bias Variance Trade-Off

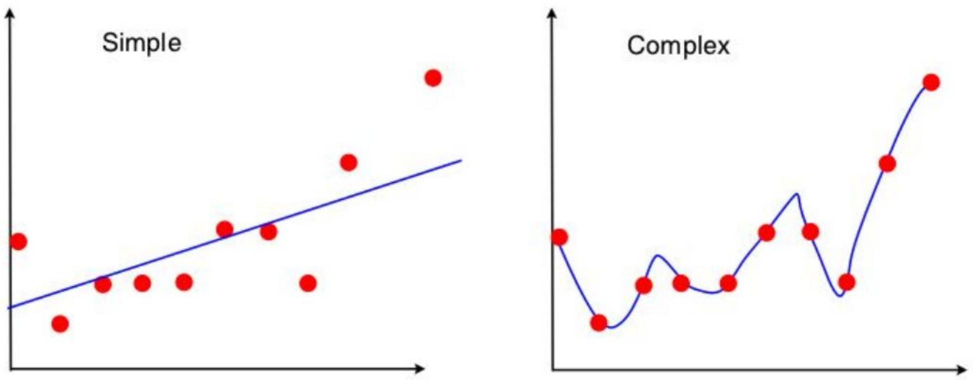
- Imagine that you can repeat your entire model by building processes to get a number of separate hits on the target.
- Each hit represents an individual realization of our model, given the chance variability in the training data we gather.

- Sometimes we will get a good distribution of training data sowe predict very well and we are close to the bulls-eye, while sometimes our training data might be full of outliers or non-standard values resulting in poorer predictions.
- These different realizations result in a scatter of hits on the target.



Bias Variance Trade-Off

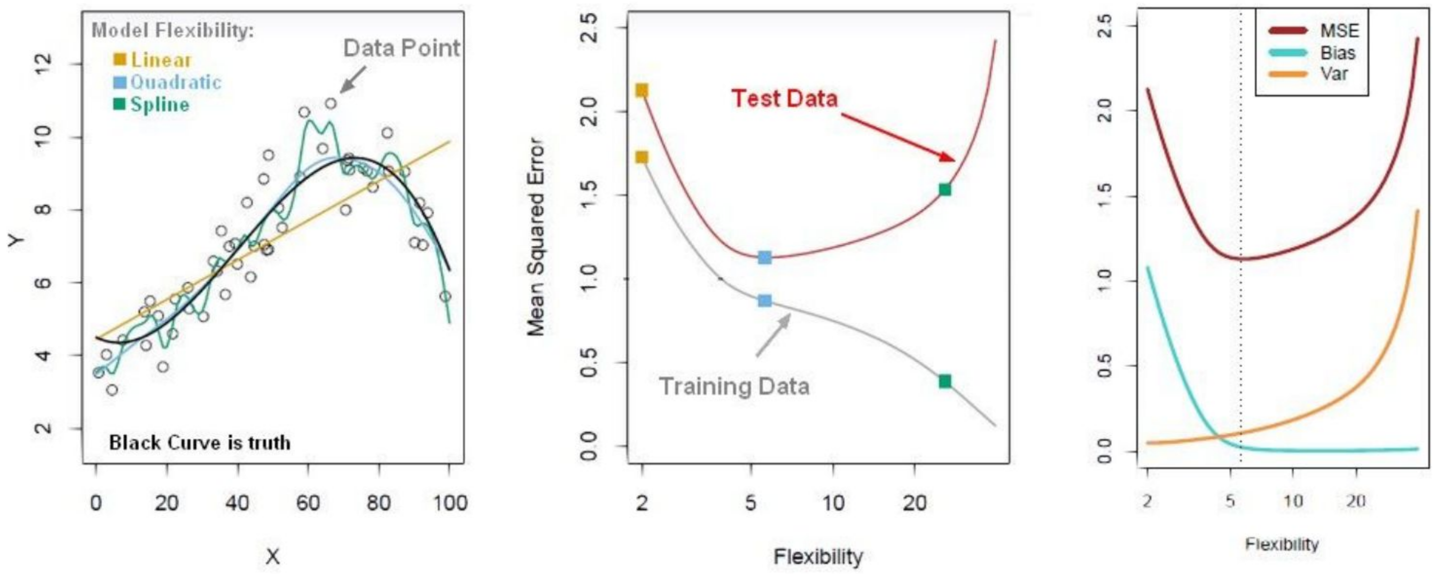
- A common temptation for beginners is to continually add complexity to a model until it fits the training set very well.



- *However, by doing this, it can cause a model to overfit to your training data and cause large errors on new data, such as the test set.*

Bias Variance Trade-Off

Below is an example model on how you can see overfitting occured from an error standpoint using test data! A black curve is used with some “noise” points on it to represent the True shape that the data follows. (source: An Introduction to Statistical Learning: With Applications in R by Daniela Witten, Gareth James, Robert Tibshirani, and Trevor Hastie)



Bias Variance Trade-Off

- Model Complexity