***Exercises for 9/14 (based heavily on textbook exercises for Chapters 1 and 2)***

*Submission instructions:*

* *Copy this document to your own drive so you can edit it.*
* ***New:*** *Rename your document with your name: FIRSTNAME.LASTNAME.02*
* *Answer each question inline (your answer should be immediately after the question). Format your answers in a color other than black so they are easy to see.*
* ***New:*** *Include a work-appropriate photo of yourself in the document (optional but appreciated)*
* *Share your document with* [*rebeccalevitan@share.brooklyn.edu*](mailto:rebeccalevitan@share.brooklyn.edu) *and* [*denys.katerenchuk@gmail.com*](mailto:denys.katerenchuk@gmail.com)*. Submission closes Thursday 9/17 at 6pm. If you have shared the document by that time, you can continue to edit it until it is graded.*

1. How does a k-nearest neighbors classifier work?

Each object is classified based on the most common class from its k (user-defined value) nearest neighbors. If most of your friends are ducks, you’re probably a duck.

1. What do model-based learning algorithms search for? What is the most common strategy they use to succeed? How do they make predictions?

Model-based algorithms look for the best model that will generalize and make accurate predictions for new data. They usually use cost functions that find how inaccurate each model is and select the best one accordingly. Once the model is selected and trained, you feed them new (unknown) data, and they base the predictions off the model parameters we previously selected.

1. What are four of the main challenges in ML?
2. Not enough training data
3. Nonrepresentative training data
4. Poor-quality data
5. Overfitting
6. What is the purpose of a labeled training set? Validation set? Test set?

Labeled training sets are to train your algorithm to recognize similarities in the features of different data points. Test sets are to test the performance of your model before putting it into use, minimizing risk of backlash. The validation set is used for fine-tuning your model to prevent overfitting and prevent over-regularizing that would reduce performance.

1. What is the difference between a model parameter and a hyperparameter?

Model parameters are used to predict information about new data, with the best generalizing ones being picked by learning algorithms. Hyperparameters are outside of the learning algorithm; they aren’t affected *by* the learning algorithm; *they* affect the learning algorithm.

1. Suppose your model performs great on the training data but generalizes poorly to new instances, what is happening? Can you name three possible solutions?

The model is overfitting to the training data. Possible solutions:

1. More training data
2. Regularization
3. Cleaning the data (reducing noise)
4. What can go wrong if you tune hyperparameters using the test set?

The hyperparameters will perform better on the test set than it would on new data, preventing the test set from doing its job - estimating real results.

1. Try different models for the housing project we did in class (the colab notebook linked from the class schedule). Fill out the table below.

NOTE: RandomizedSearchCV was used to find ideal hyperparameters for model 5.

|  |  |  |  |
| --- | --- | --- | --- |
| **Model** | **Training error** | **Validation error** | **Hyperparameters** |
| KNeighborsRegressor | 44310 | 63617 |  |
| AdaBoostRegressor | 80817 | 82243 |  |
| SVR | 111094 | 111809 |  |
| VotingRegressor (using models 1 & 3) | 68648 | 77021 |  |
| RandomForestRegressor | 19178 | 49933 | {'n\_estimators': 30, 'max\_features': 8} |