**Chapter 4: Tree-based models**

Theoretical part:

1. Decision Trees:

Read chapter 9 in ESL, pages 305-313.

Explain to yourself the following, then discuss it with your supervisor.

1. How might one use a classification tree to return class probabilities?
2. What are the three measures used for building classification trees? What are their relative merits? What’s used for regression trees?
3. How can the Gini index be interpreted?
4. What are the disadvantages of decision trees?
5. What does pruning aim to solve? How is it done?
6. What might be a problem with classification decision trees when a categorical feature with many possible values? Suggest a solution to the problem.
7. Random Forest:

Read ISLR p.315-321.

1. What is the idea of bagging?

2. What is the trade off in choosing the number m of features to sample in every node?

3. What’s Out Of Bag Sampling? What is the size of the Out Of Bag set when the dataset is large?

4. How can one measure feature importance in random forest?

Now read about random forest in chapter 15 of ESL, pages 587-596. Skip 15.3.3 (Proximity Plots).

5. Solve exercise 15.1. What is its implication on random forest?

6. Demonstrate in code that when the number of variables is large, but the fraction of relevant variables is small, random forests are likely to perform poorly with small m.

1. Isolation Forest:

* Read the intuitive explanation in the following blogpost:

<https://quantdare.com/isolation-forest-algorithm/>

* And then a more profound reading here: <https://towardsdatascience.com/outlier-detection-with-isolation-forest-3d190448d45e>

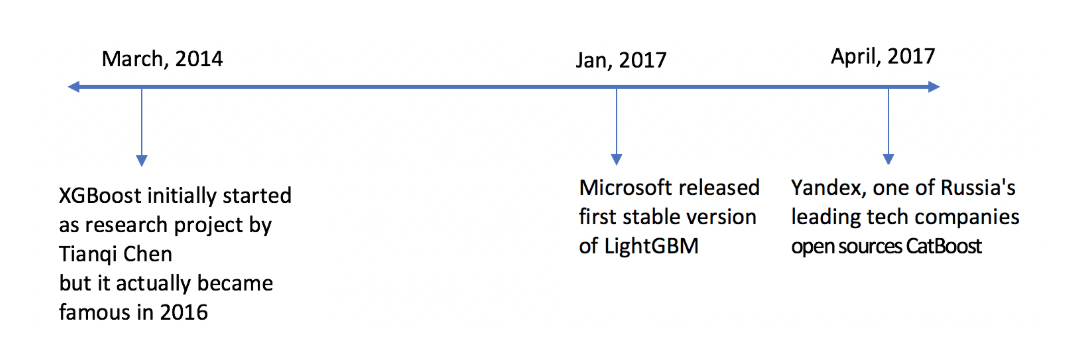
1. Boosting methods:

Read the following two blog posts :

* The difference between bagging and boosting <https://medium.com/mlreview/gradient-boosting-from-scratch-1e317ae4587d>
* Deep explanation of adaboost and comparison to standard gradient boosting:<https://towardsdatascience.com/boosting-algorithms-explained-d38f56ef3f30>

1. Advanced Boosting methods:

First pay attention to the order of publishment of the three articles:



And now we go to the reading! (Look forward to hearing from your post ...)

* Mathematical background to gradient boosting: <https://towardsdatascience.com/the-good-old-gradient-boosting-f4614b0e62b0>
* Read the next blogpost on xgboost (in order to get an overview of the algorithm, later you will read the article itself)

<HTTPS://TOWARDSDATASCIENCE.COM/XGBOOST-B736C2AB10CE>

* Read the article of xgboost (attached in the folder) – focus on the classification part rather than the calculations (no need to read about distributed computations) but understand the general concepts that were used and where it is possible to improve performance.
* LightGBM introduction <https://towardsdatascience.com/lightgbm-800340f21415>

In the parts where you feel that they do not explain all the details (for example in the EFB section) you have an article (attached in the folder) and try to understand from there. If still the subject is unclear try searching other posts on the Internet.

* Read the blogpost of Tal Peretz (who founded this team) on catboost <https://towardsdatascience.com/https-medium-com-talperetz24-mastering-the-new-generation-of-gradient-boosting-db04062a7ea2>

Here too we think that the main point of the article - ordered boosting, is relatively hard to understand it is better to understand from the pseudo code in the article itself.

Practical part:

In this exercise, we will dive into your first kaggle exercise . While you have read and seen the different tree models as a classification algorithm, they can also be activated as regression one, and they are even quite common.

The next kaggle:<https://www.kaggle.com/c/new-york-city-taxi-fare-prediction/overview>

Describes taxi prices in New York, containing columns describing the trip, and requesting that you produce a tariff contract.

You must build a classifier with maximum accuracy to predict the aforementioned.

In your troubleshooting you must include the following steps:

* Aksploratia of the information.
* Production of new features that can help with the model's work.
* Testing models (trees and other methods you have learned)
* Compare the models that were examined.
* Conclusions.