ML cheat sheet

1. **Terms:**

* **Numpy** – python package for linear algebra(matrix and matrix functions, other math functions)

ex: numpy array

* **Pandas** – data processing package that allows data manipulation

ex: pd.read\_csv

* **scikit-learn** – package for ML with classification, regression, clustering algos.
* **Logistic regression** - gives the true/false result. Predicts 2 discreet values, fits an S
* **Linear regression** – predicts continuous values, fits a line. We need a threshold to classify because of S fit.
* **Random forest** – multiple decision trees created from random sets from training set and random selected features. The prediction is done by running all trees and the most votes win.
* **Naïve bayes** P(H|E) = [P(H) \* P(E|H)] / P(E)
* **Residual** – distance from the line to the point
* **R2**  - sum of the squared residuals

**R2** can tell us how much of the of the variable variance is dependent of the other variable, need to be large

**R2** = [SS(mean)-SS(fit)] / SS(mean) = x %

* **Least squares** – helps fitting a line in linear regression, the smallest residuals squared
* **Variance** – R2/sample size
* **P-value** – determines if the R2 is of any statistical significance, needs to be small
* **Cross-validation:** protests against overfitting, k-fold CV is where the training data is split in samples(folds) and validated against the other folds.
* **Loss-function** – evaluates an algo
* **Mean square error** is measured as the average of squared difference between predictions and actual observations. – regression
* **Cross Entropy Loss** increases as the predicted probability diverges from the actual label. - classification

1. **Titanic flow:**
2. What is the type of our data? Is it images, tabular data, text, time series data, ... ?

tabular

1. What is the type of the problem, i.e. is it regression, classification, clustering, or something different?

classification

1. What is the metric used for the final evaluation?

Accuracy = (**TP**+**TN**)/(**TP**+**TN**+**FP**+**FN**)

a. Load train data – csv with passenger data

b. Inspect data to understand the input(EDA): features,target and size. Eliminate features that clearly don’t tell anything.

c. Check missing/incomplete data in the dataset

1. Try not to ditch missing data for one missing feature

2. We can use the values from higly correlated features. For age we can put class if age and class are correlated(heatmap), mean.

d. Train/test split or load test data.

Fit the train data into a model(Linear validation)

e. Feature engineering: **the transformation of data into features, with the aim of representing better the problem we would like to solve, and resulting to a better model performance on unseen data.**

**f.**