Top 10 - on DATA

1. **Check and understand input data** and its most basic relations.

2. **Split data (Train, Valid, and Test),** to not (unknowingly) overtrain.

3. **Consider number of features**, and omit those which don't contribute.

4. **Ensure LOTS OF useful quality data**, either by having it or getting it.

5. **Fast computing and access to data** for quick analysis turn-around is key.

6. **If data is in shortage,** consider methods for augmenting.

7. **All data is flawed**. Make sure you know how and filter it first.

8. **Map out any missing data** and decide wether to filter or impute.

9. **Test several methods**, as different methods apply to different cases.

10. **Make cross checks**. All data analysis results can be bugged/flawed.

Top 10 - on ALGORTIHMS

1. **Use appropriate model/architecture** that matches the data/problem.
2. **Think carefully about the loss function**, i.e. what you want to optimise.
3. **Tree algorithms are good for getting fast results** on <u>structured</u> data.
4. **Neural networks are more performant and versatile**, but harder to train.
5. **Variable transformation** is typically required for Neural Networks.
6. **Image analysis** is mostly done with a Convolutional Neural Network.
7. **Dimensionality reduction** benefits very high dimensional problems.
8. **Streams of data** (e.g. text) can be analysed with LSTM/RNN networks.
9. **Unsupervised learning/clustering** results can be difficult to interpret.
10. **Uncertainties in regression** can be given by ML (typically NN) algorithms.