Machine Learning Checklist

The Eight Main Steps:

1. Frame the problem and look at the big picture.
2. Get the data.
3. Explore the data to gain insights.
4. Prepare the data to better expose the underlying data patterns to ML algorithms.
5. Explore many different models and shortlist the best ones.
6. Fine-tune your models and combine them into a great solution.
7. Present your solution.
8. Launch, monitor, and maintain your system.

Tips:

- Automate as much as possible (without overdoing it).
- Always keep the original, raw data intact.
- Write functions for all data transformations that are applied.

Frame the problem and look at the big picture.

☐ Define the objective in business terms.

☐ How will your solution be used?

☐ What are the current solutions/workarounds (if any)?

☐ How should you frame this problem (supervised/unsupervised, online/offline, etc.)?

☐ How should performance be measured?

☐ Is the performance measure aligned with the business objective?

☐ What would be the minimum performance needed to reach the business objective?

☐ What are comparable problems? Can you re-use experience or tools?

☐ Is human expertise available?

☐ How would you solve the problem manually?

☐ List the assumptions that you (or others) have made so far?

☐ Verify assumptions if possible.

Get the data.

☐ List the data you need and how much you need.

☐ Find and document where you can get that data.

☐ Check how much space it will take.

☐ Check legal obligations and get authorization if necessary.

☐ Get access authorizations.

☐ Create a workspace (with enough storage space).

☐ Convert the data to a format you can easily manipulate (without changing the data itself).

☐ Ensure sensitive information is deleted or protected.

☐ Check the size and type of data (time series, geographical, etc.).

☐ Sample a test set, put it aside, and never look at it.

Explore the data to gain insights.

☐ Create a copy of the data for exploration (sampling it down to a manageable size if necessary).

☐ Create a Jupyter notebook to keep a record of your data exploration.

☐ Study each attribute and it's characteristics, including: name, type, % of missing values, noisiness and type of noise, usefulness for task, type of distribution.

☐ For supervised learning tasks, identify the target attribute(s).

☐ Visualize the data.

☐ Study the correlations between attributes.

☐ Study how you solve the problem manually.

☐ Identify the promising transformations you may want to apply.

☐ Identify extra data that would be useful (go back to the "Get the Data" section).

☐ Document what you've learned.

Prepare the data to better expose the underlying data patterns to ML algorithms.

☐ Clean the data:

  ☐ Fix or remove outliers

  ☐ Fill in missing values or drop their rows.

☐ Perform feature selection:

  ☐ Drop the attributes that provide no useful information for the task

☐ Perform feature engineering:

  ☐ Discretize continuous features.

  ☐ Decompose features.

☐ Add promising transformations of features.

  ☐ Aggregate features into promising new features.

☐ Perform feature scaling:

  ☐ Standardize or normalize features

Explore many different models and shortlist the best ones.

☐ Train many quick-and-dirty models from different categories (linear, naive Bayes, SVM, random forest, NN) using standard parameters.

☐ Measure and compare their performance with K-Fold cross validation.

☐ Analyze the most significant variables for each algorithm.

☐ Analyze the types of errors the models make.

☐ Perform a quick round of feature selection and engineering.

☐ Perform one or two more quick iterations of the five previous steps.

☐ Shortlist the top 3-5 most promising models, preferring models that make different types of errors.

Fine-tune your models and combine them into a great solution.

☐ Fine-tune the hyperparameters using cross validation:

☐ Treat your data transformation choices as hyperparameters, especially when you are not sure about them.

☐ Unless there are very few hyperparameter values to explore, always go Bayesian, random, grid search.

☐ Try ensemble methods. Combining your best models will often produce better performance than running them individually.

☐ Once you are confident about your final model, measure its performance on the test set to estimate the generalization error.

Present your solution.

☐ Document what you have done.

☐ Create a nice presentation: Make sure you highlight the big picture first. Explain why your solution achieves the business objective.

☐ Present interesting points you noticed along the way. Describe what worked and what did not.

☐ List your assumptions and your system's limitations.

☐ Communicate key findings through beautiful visualizations or easy-to-remember statements.

Launch, monitor, and maintain your system.

☐ Get your solution ready for production:

☐ Plug into production data inputs.

☐ Write unit tests.

☐ Write monitoring code to check your systems live performance at regular intervals, and trigger alerts when it drops:

☐ Be aware of slow degradation/model rot.

☐ Monitor your input quality.

☐ Retrain your models on a regular basis on fresh data.