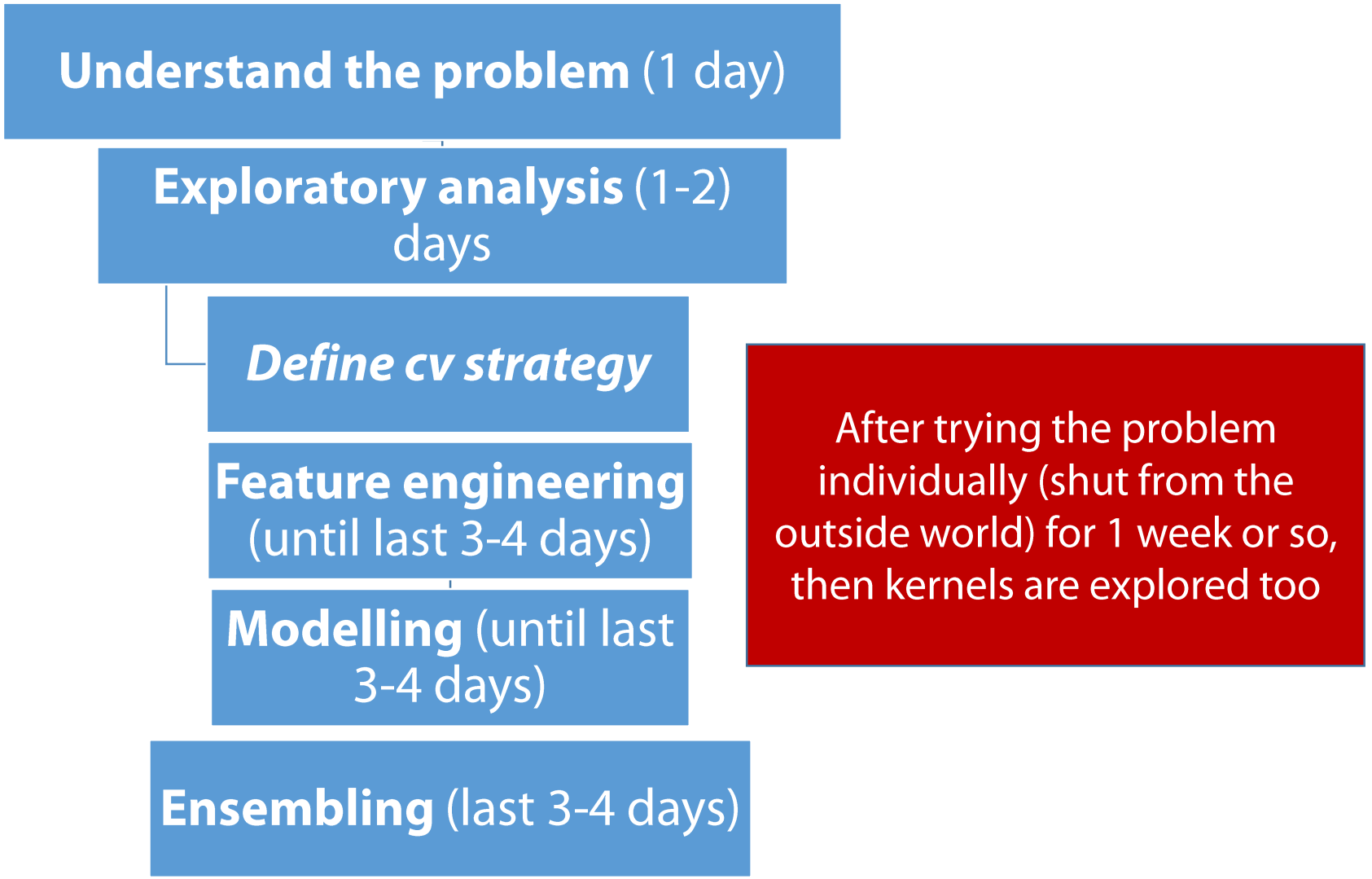
Competition Pipeline

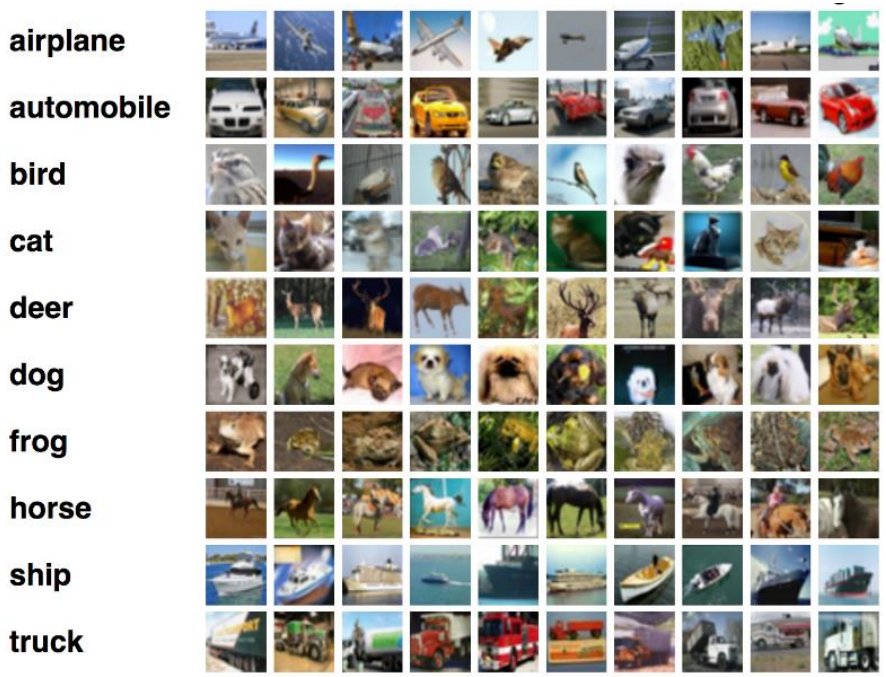
By Marios Michailidis

The Pipeline



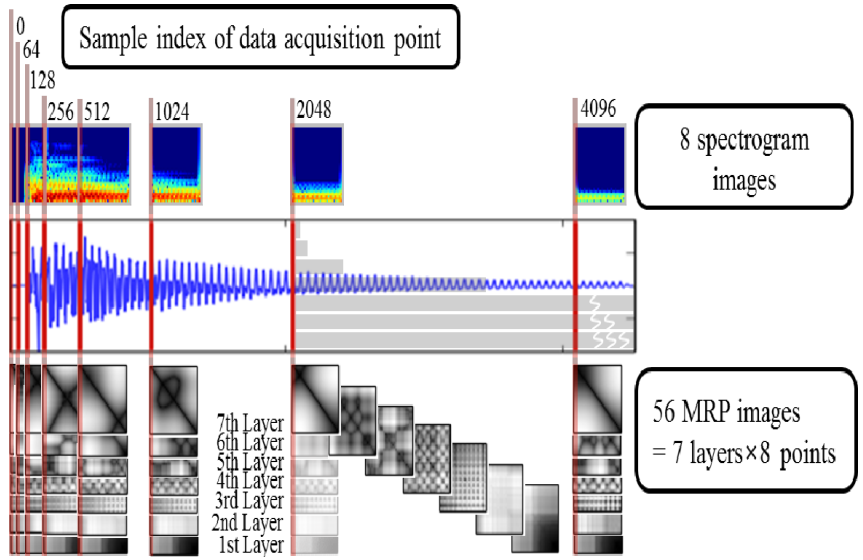
Understand broadly the problem

- Type of problem



Understand broadly the problem

- Type of problem

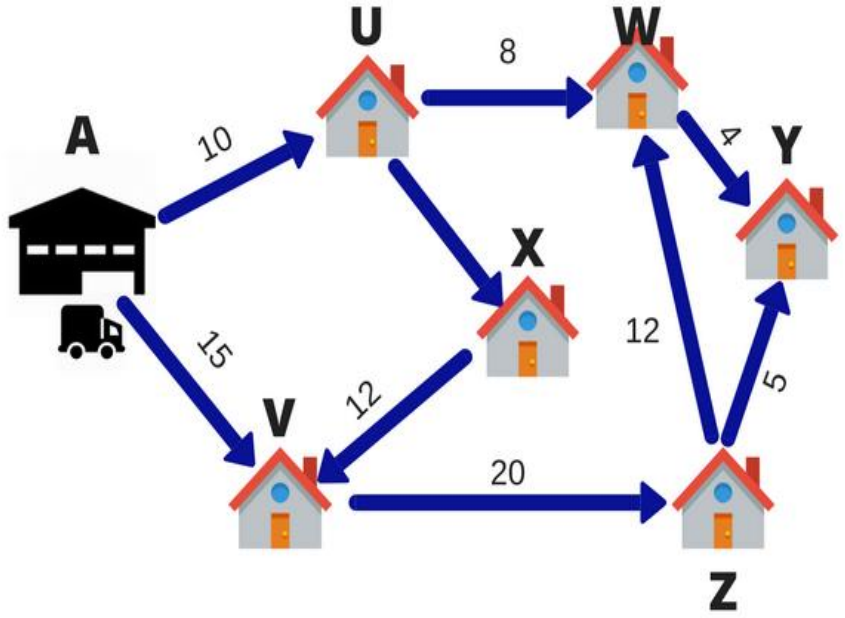


Understand broadly the problem

- Type of problem

Understand broadly the problem

- Type of problem



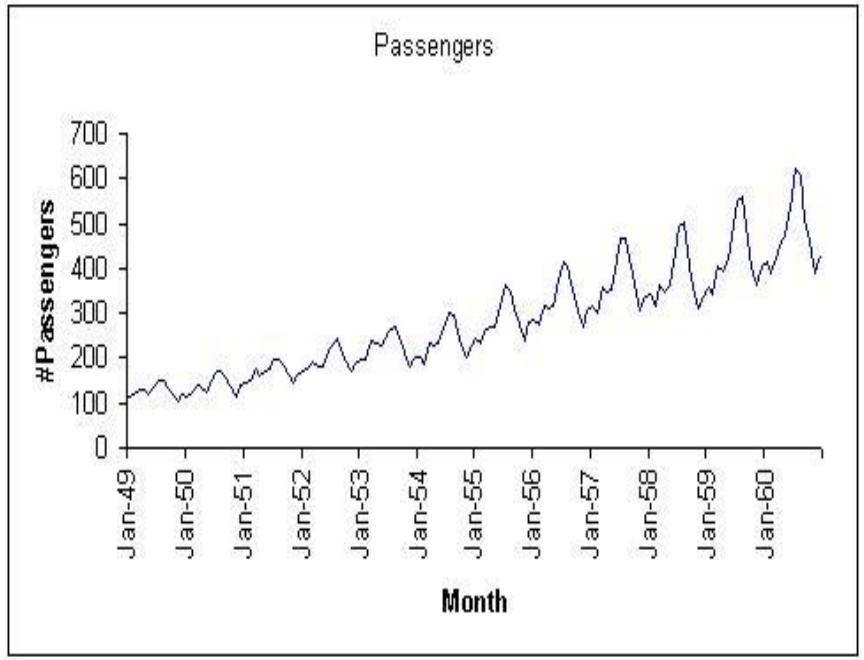
Understand broadly the problem

- Type of problem

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
A	91	558	96	113	109	125	57	109	115	102	92	89	42	51	45
B	21	1	21	21	18	18	8	25	17	20	16	21	11	8	10
C	75	652	93	232	290	278	291	332	99	109	101	97	54	50	48
D	100	19	122	70	97	90	41	110	94	85	97	98	60	46	47
E	97	22	122	86	112	102	47	104	81	898	79	101	55	54	47
F	83	23	103	96	114	483	46	95	97	94	93	92	50	51	52
G	12	2	10	10	8	10	0	8	10	8	9	9	6	5	247
H	96	20	109	102	106	107	48	117	98	88	94	111	44	46	48
I	100	21	98	87	97	95	45	92	119	111	86	106	50	57	49
J	101	18	81	83	777	111	52	115	100	100	109	99	44	48	45
K	96	19	96	97	400	105	43	86	103	112	92	86	52	52	48
L	104	22	100	940	120	109	52	958	112	116	92	96	49	50	45
M	45	9	49	42	280	47	28	50	47	40	49	50	25	22	29
N	81	16	80	1159	78	77	38	773	66	73	64	80	35	41	37
O	8	2	8	9	9	12	312	11	9	9	9	8	4	6	5

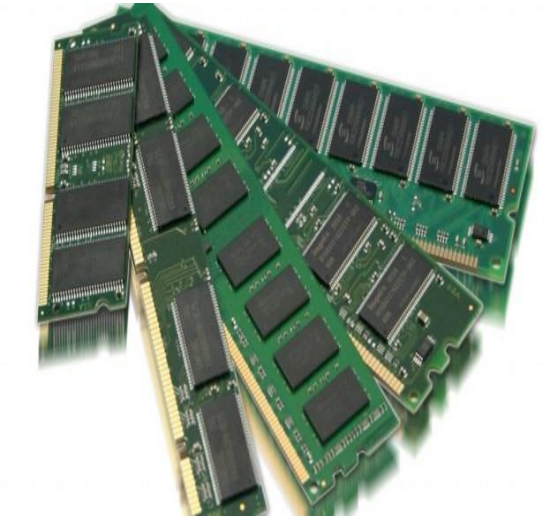
Understand broadly the problem

- Type of problem



Understand broadly the problem

- Type of problem
- How BIG is the data (How much I need?)?
- Hardware needed (CPUs, GPUs, RAM, Disk space)



Understand broadly the problem

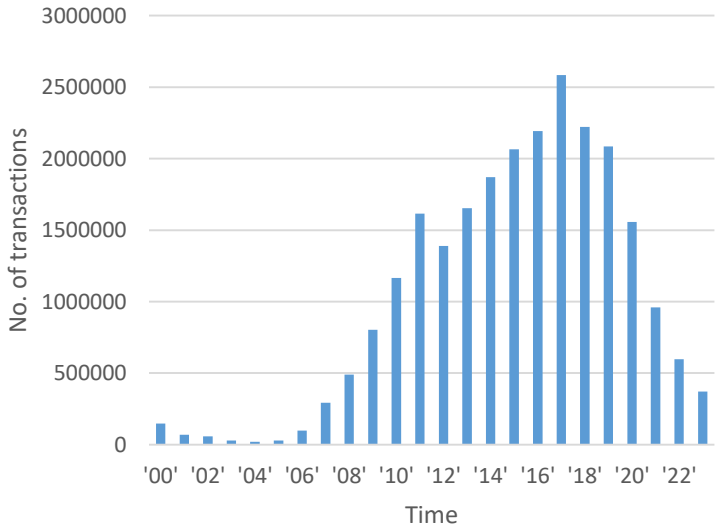
- Type of problem
- How BIG is the data (How much I need?)?
- Hardware needed (CPUs, GPUs, RAM, Disk space)
- Software Needed (TF, sklearn, Lightgbm, xgboost)

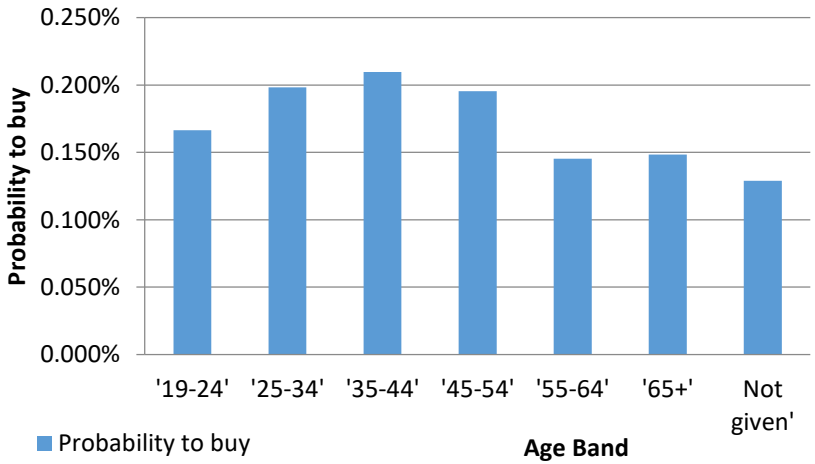
Understand broadly the problem

- Type of problem
- How BIG is the data (How much I need?)?
- Hardware needed (CPUs, GPUs, RAM, Disk space)
- Software Needed (TF, sklearn, Lightgbm, xgboost)
- What is the metric being tested on?
- Previous code relevant?

Do some (exploratory data analysis) EDA

- Plot histograms of variables. Check that a feature looks similar between train and test.
- Plot features versus the target variable and vs time.
- Consider univariate predictability metrics (IV,R,auc)
- Binning numerical features and correlation matrices





Decide a cross validation Strategy

- This step is critical. Its success is a good indication for what is going to happen in the competition.
- *People have won by just selecting the right way to validate.*
- Is time is important? Split by time. **Time-based validation**.
- Different entities than the train. **Stratified validation.**
- Is it completely random. **Random validation** (random K-fold).
- **Combination** of all the above.
- Use test leader board to test.

Feature engineering

- The type of problem defines the feature engineering.
- **Image classification**: Scaling, shifting, rotations, CNNs. Suggestion previous data science bowls.
- **Sound classifications**: Fourier , Mfcc, specgrams, scaling **.** Tenso flow speech recognition
- **Text classification**: Tf-idf, svd, stemming, spell checking, stop words' removal, x-grams. StumbleUpon Evergreen Classification.
- **Time series**: Lags, weighted averaging, exponential smoothing . Walmart recruitment.
- **Categorical** : Target enc, freq, one-hot, ordinal, label encoding. Amazon employee
- **Numerical** : Scaling , binning, derivatives ,outlier removals, dimensionality reduction. Africa soil.
- **Interactions**: multiplications, divisions, group-by features . Concatenations. Homesite.
- **Recommenders**: Features on transactional history. Item popularity,

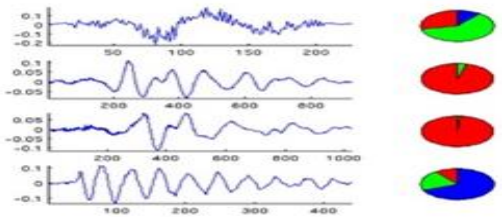
frequency of purchase. Acquire Valued Shoppers.

- This process **can be automated** using selection with cross validation.

Feature engineering

Scaling, Shifts, Rotations

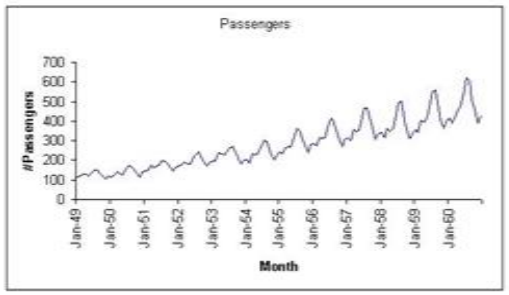
Filters,Flips...



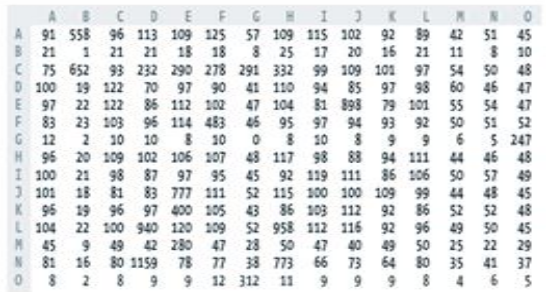
MFCCs,Specgrams...

TF-IDF, Stemming, Spellcheck

Stopwords, SVD, Word2vec...



Lags, Exponential Smoothing



Categorical Encoding

Numerical Encoding

Item popularity, FMs

Frequency of Purchase

Different problems require different feature engineering

Can be automated

Modeling

- The type of problem defines the feature engineering.
- **Image classification**: CNNs (Resnet, VGG, densenet…)
- **Sound classifications**: CNNs(CRNN), LSTM
- **Text classification**: GBMs, Linear, DL, Naïve bayes, KNNs, LibFM, LIBFFM
- **Time series**: Autoregressive models, ARIMA, linear, GBMs, DL, LSTMs
- **Categorical features**: GBMs, Linear models, DL , LibFM, libFFm
- **Numerical Features**: GBMs, Linear models, DL, SVMs
- **Interactions**: GBMs, Linear models, DL
- **Recommenders**: CF, DL, LibFM, LIBFFM, GBMs
- Each tuned individually. Different datasets. Bagged

Ensembling

- All this time, **predictions** on internal validation and test **are saved.**

(If **collaborating** with others, this is the point where everyone passes on their predictions as .csv files)

- Different ways to combine from **averaging** to multilayer **stacking.**
- Small data requires simpler ensemble techniques (like averaging).
- Helps to average a few **low-correlated predictions** with good scores.
- Bigger data can utilize stacking.
- **Stacking process repeats** the modelling process.

Tips on collaboration

- It makes it more fun.
- You learn more.
- You score better.
- You gain in at least 2 ways. First you can cover more ground. Second every person seizes the problem from different angles leading to more thorough solutions.
- Start collaborating after getting some experience (maybe 2-3 competitions) to understand the dynamics.
- Start with people around your “rank”.
- Look for people that are likely to do different things

well or that specialize in certain areas.

Selection final submissions

- Normally select the **best submission locally and best on leader board**.
- It is good to **monitor correlations**. If correlations are too high and submissions exist with high scores but significantly lower correlations, they could be considered too.

Final tips

- In these challenges **you never lose**. You may not win prize money, BUT you always gain in terms of knowledge, experience, meeting/collaborating with talented people in the field, boost your CV.
- Coffee is kind of a must when you do this!
- See it **like a game**…you have some tools …or "weapons" , you can get a score and you try to beat the "bad guys"' score!
- **Take a break** often to rest your mind – do some physical exercise as it is unhealthy sitting on chair many hours.
- The kaggle community may be the most kind, helpful

community I have ever experienced in any social context.

- After the competition look for people sharing approaches.
- Create a notebook with useful methods and update it.