Empirical Evaluation

F21RP - Research Methods and Project Planning

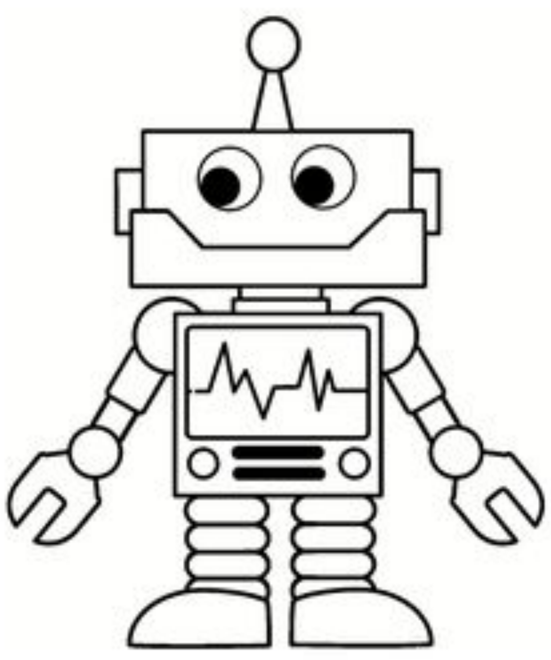
(based on slides by Oliver Lemon, Judy Robertson, Helen Hastie, Frank Broz and Tom Methven)

Learning Outcomes

- Datasets
  - Train/Dev/Test splits
- Metrics
- Reporting Results
  - Ablation Study
  - Discussion of Results
  - Error Analysis

Automatic Evaluation

- In case of Machine Learning / Data Science projects
  - Usually there is an existing labelled dataset
    - Learn a model
  - We can automatically compute performance of model by measuring how well it can predict labels on an unseen test set



An example dataset

Features

Feature Values

Easy?	AI?	Systems?	Theory?	Morning?	Rating	Label
y	y	n	y	n	+2	like
y	y	n	y	n	+2	like
n	y	n	n	n	+2	like
n	n	n	y	n	+2	like
n	y	y	n	y	+2	like
y	y	n	n	n	+1	like
y	y	n	y	n	+1	like
n	y	n	y	n	+1	like
n	n	n	n	y	0	like
y	n	n	y	y	0	like
n	y	n	y	n	0	like
y	y	y	y	y	0	like
y	y	y	n	y	-1	not like
n	n	y	y	n	-1	not like
n	n	y	n	y	-1	not like
y	n	y	n	y	-1	not like
n	n	y	y	n	-2	not like
n	y	y	n	y	-2	not like
y	n	y	n	n	-2	not like
y	n	y	n	y	-2	not like

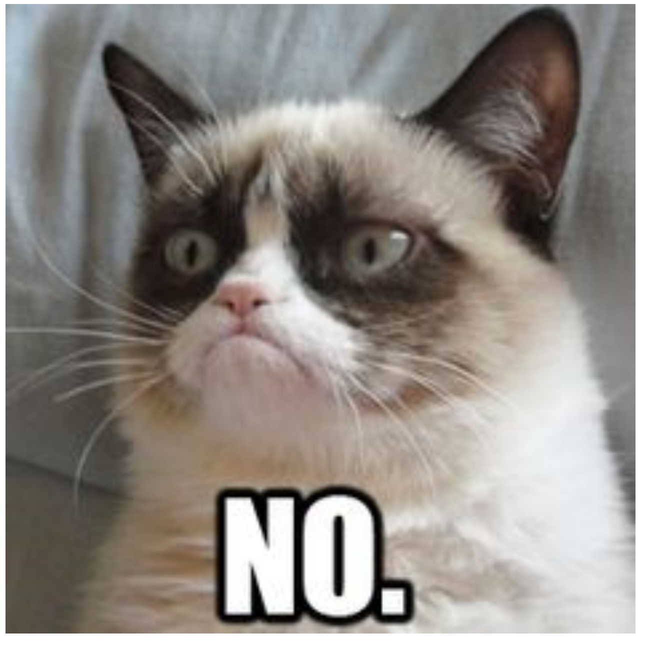
Train/Dev/Test sets

In practice, we always split examples into 3 distinct sets:

- Training set
  - Used to learn the parameters of the ML model
  - e.g., what are the nodes and branches of a decision tree
- Development set
  - aka tuning set, or validation set, or held-out data
  - Used to learn hyperparameters
    - Parameters that control other parameters of the model
    - e.g., max depth of decision tree, or regularisation term λ
- Test set
  - Used to evaluate how well we're doing on new unseen examples

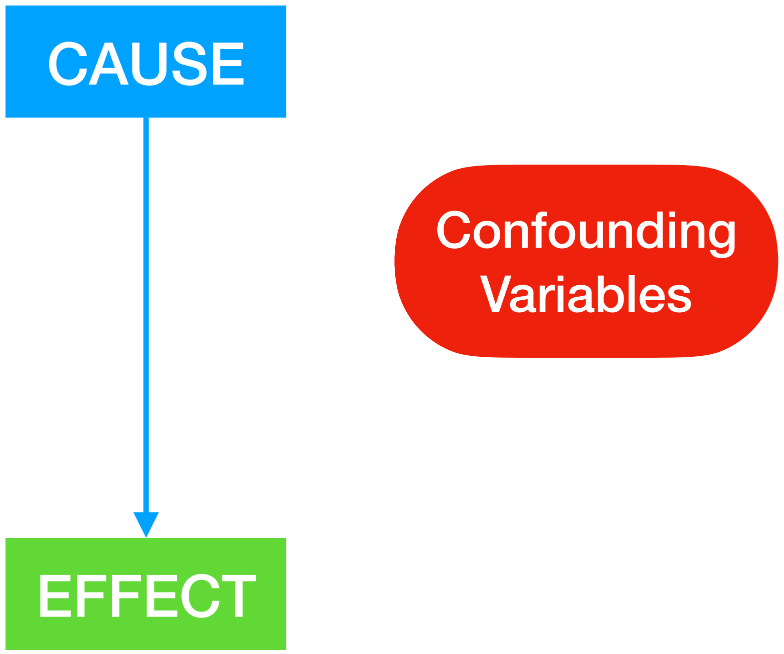
Cardinal Rule of Machine Learning

Never ever touch your test data!



Independent/Dependent Variables (Automatic Evaluation)

- Independent Variable
  - Variable that is manipulated
    - e.g., type of training loss, new features, architecture of the model
- Dependent Variable
  - Variable that is measured
    - e.g., F-1 score



Metrics

Computing Accuracy

- Accuracy: % of predictions that are correct
  - $Accuracy = \dfrac{True\ Predicted\ Labels}{All\ Labels}$
- Example (Census Income)
  - Labels: “Income >$50k/yr”, “Income <$50k/yr”
  - Test set size: 13300 examples
  - trained SVM model predicts 11125 examples, correctly
  - $Accuracy = \dfrac{11125}{13300} = 83.65\,\%$

Re-evaluating Accuracy

- Accuracy not always appropriate for classification
  - Some errors matter more than others
    - Cancer detection
    - Spam email
    - In general: X-vs-not-X
  - Imbalanced datasets
    - (Census Income) What if “Income <$50k/yr” label appears 12700 times in the dataset?
    - High accuracy probably comes because of majority label

Precision/Recall

- Example (Spam Detection)
  - +1 means spam, -1 means ham
- Categorise predictions using confusion matrix
  - True/False Positives
  - True/False Negatives

	Gold Label (+1)	Gold Label (-1)
Prediction (+1)	TP	FP
Prediction (-1)	FN	TN

Precision/Recall

- Example (Spam Detection)
  - +1 means spam, -1 means ham
- Precision: % of Positive predictions that are correct
  - $Precision = \frac{TP}{TP + FP}$
- Recall: % of Positive gold labels that are predicted
  - $Recall = \frac{TP}{TP + FN}$

	Gold Label (+1)	Gold Label (-1)
Prediction (+1)	TP	FP
Prediction (-1)	FN	TN

F-Measure (F-score)

$$F_\beta = \frac{(1+\beta^2) \times Pr \times Rec}{(\beta^2 \cdot Pr) + Rec}$$

- Harmonic mean of Precision and Recall
  - Favours systems with equal Precision and Recall
  - Imbalanced scores: F-score drops dramatically
  - Usually $\beta = 1$

Task-specific metrics

- Sometimes tasks have their own measurements of performance
  - Composite F-1 score
    - Average of F-1 scores for several sub-tasks
    - E.g., Semantic Role Labeling “score” consists of F-1 scores for 4 different sub-tasks
  - Precision or recall-oriented metrics with heuristics
    - BLEU / METEOR (Machine Translation), ROUGE (Summarisation)
  - Error rates
    - Word Error Rate (Automatic Speech Recognition), Sentence Error Rate
  - Information Retrieval
    - E.g., Recall@k, HITS@k, NDCG
- Make sure you specify which metrics you are going to use in your report!

Cross Validation

- So far we have used a development set to perform hyperparameter tuning
  - Waste part of training data (esp. if we have few hyperparameters)
- Cross-val:
  - Split training set into K equally-sized partitions
  - Use K-1 for training, and Kth for testing
  - Repeat process for K times
  - Average F-score/Accuracy
- Typically K=5, 10
- Pros: Robust
- Cons: Slow

Results: Ablation Study

- Show incremental decrease in DV (usually performance) when progressively removing (ablating) IVs between consecutive experiments
- More common using automatic metrics (faster/cheaper)
- **Note: Always do the ablations on the dev set; never ever on the test set!**
- Example
    - Machine Translation system from French to English
    - IVs:
        - Feature 1 (back-translation language model)
        - Feature 2 (context gate)
        - Feature 3 (lexical coherence model)
    - DV: BLEU score, METEOR score

Models	BLEU	METEOR
Full Model (BackLM+CtxG+Coh)	33.21	45.2
BackLM+CtxG	30.10	43.1
BackLM	25.3	36.24

Reporting Results

- Usually present table(s) of results per experiment containing all metrics
  - Sometimes we split to multiple tables for clarity's sake
- Include if possible multiple "simple" baselines, and as many state-of-the-art models as possible
- Discuss results!
- Make critical comparisons and give possible explanations
  - "As seen in Table 1, our proposed model beats the baseline by a margin of 1.8 BLEU score. Incorporating our novel *context gate* feature seems to be crucial."
- Use ablation study to justify your explanations
  - "[…] This is further supported by our ablation study (Table 2) which shows a substantial decrease in performance when removing the context gate feature"

Models	BLEU	METEOR
PBMT Baseline	27.3	38.2
seq2seq	28.5	40.9
seq2seq w/ attention	29	41.6
Transformer	31.5	42.7
CNN-based	31.3	42.5
Full Model	**32.1**	**44.4**

Table 1 (Results on test set)

Models	BLEU	METEOR
Full Model (BackLM+CtxG+Coh)	33.21	45.2
BackLM+CtxG	30.10	43.1
BackLM	25.3	36.24

Table 2 (Ablation Results on dev set)

Results: Error Analysis

- (Often omitted) Analyse the mistakes your system is making on a small portion of your test set
  - Randomly select 10-50 examples with a misclassification or poor performance
- Two-fold methods (either/or both):
  - Check input conditions (usually for classification tasks)
    - e.g., abnormally high value of a feature, repetitive value of a feature triggers
  - Check output (usually for tasks that generate a structured output)
    - e.g., wrong order of words in translation (adjective order wrong
- Qualitative analysis
  - Simply give explanations for mistakes using examples
  - Also include a table with frequencies of “made-up” classes of error

Error	Percent (%)
Negation	12%
Wrong Order	30%
Wrong syntax	55%
Wrong lexical	8%

Attributions

- https://static.makeuseof.com/wp-content/uploads/2017/04/experiment_lab-670x447.jpg
- https://thepolymathproject.com/wp-content/uploads/2018/07/steel-man.png
- https://webgnomes-webgnomesllc.netdna-ssl.com/wp-content/uploads/2012/07/seo-analysis.jpg
- https://i.pinimg.com/originals/d7/ff/6c/d7ff6cfc680189370e446d2a7f18c09b.jpg
- https://image.shutterstock.com/image-illustration/abc-building-blocks-on-white-260nw-109234082.jpg
- https://www.dictionary.com/e/wp-content/uploads/2018/04/hmm.jpg
- https://i.pinimg.com/236x/d7/a3/ae/d7a3ae5506817d1ef60dabde37150fe9--grumpy-cat-humor-grumpy-cats.jpg
- http://clipart-library.com/clipart/832162.htm