

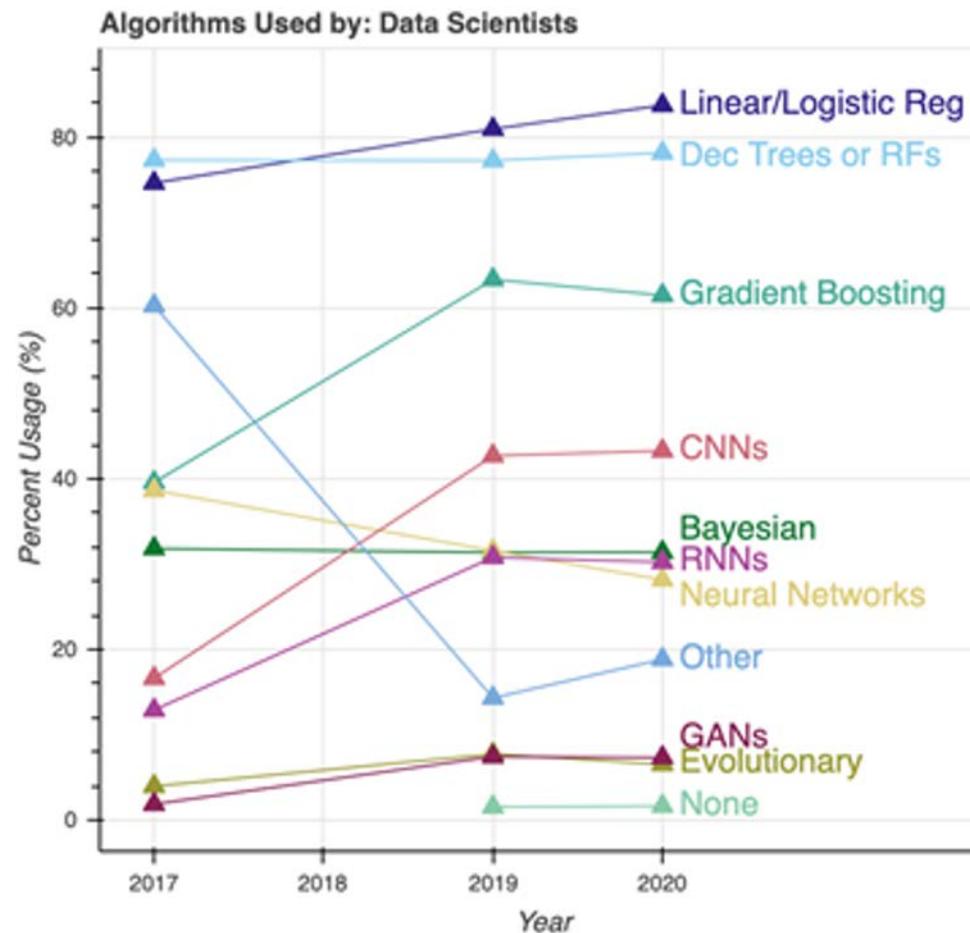
TRICKS OF THE TRADE:

Machine learning in the real world

Burning questions

- Which model to use?
- How should I improve my models?
 - Diagnosis and error analysis

What to use



<https://medium.com/analytics-vidhya/ongoing-kaggle-survey-picks-the-topmost-data-science-trends-7c19ec7606a1>

What's the best?

We evaluate **179 classifiers** arising from **17 families** (discriminant analysis, Bayesian, neural networks, support vector machines, decision trees, rule-based classifiers, boosting, bagging, stacking, random forests and other ensembles, generalized linear models, nearest-neighbors, partial least squares and principal component regression, logistic and multinomial regression, multiple adaptive regression splines and other methods), implemented in Weka, R (with and without the caret package), C and Matlab, including all the relevant classifiers available today. We use **121 data sets**, which represent the whole UCI data base (excluding the large-scale problems) and other own real problems, in order to achieve significant conclusions about the classifier behavior, not dependent on the data set collection. The classifiers most likely to be the bests are the random forest (RF) versions, the best of which (implemented in R and accessed via caret) achieves 94.1% of the maximum accuracy overcoming 90% in the 84.3% of the data sets. However, the difference is not statistically significant with the second best, the SVM with Gaussian kernel implemented in C using LibSVM, which achieves 92.3% of the maximum accuracy. A few models are clearly better than the remaining ones: random forest, SVM with Gaussian and polynomial kernels, extreme learning machine with Gaussian kernel, C5.0 and avNNet (a committee of multi-layer perceptrons implemented in R with the caret package). The random forest is clearly the best family of classifiers (3 out of 5 bests classifiers are RF), followed by SVM (4 classifiers in the top-10), neural networks and boosting ensembles (5 and 3 members in the top-20, respectively).

Most data sets are small (<1000) in this paper

<http://jmlr.org/papers/volume15/delgado14a/delgado14a.pdf>

Revisiting Deep Learning Models for Tabular Data

Yury Gorishniy^{*†‡}

Ivan Rubachev^{†♣}

Valentin Khrulkov[†]

Artem Babenko^{†♣}

† Yandex, Russia

‡ Moscow Institute of Physics and Technology, Russia

♣ National Research University Higher School of Economics, Russia

Abstract

The existing literature on deep learning for tabular data proposes a wide range of novel architectures and reports competitive results on various datasets. However, the proposed models are usually not properly compared to each other and existing works often use different benchmarks and experiment protocols. As a result, it is unclear for both researchers and practitioners what models perform best. Additionally, the field still lacks effective baselines, that is, the easy-to-use models that provide competitive performance across different problems.

In this work, we perform an overview of the main families of DL architectures for tabular data and raise the bar of baselines in tabular DL by identifying two simple and powerful deep architectures. The first one is a ResNet-like architecture which turns out to be a strong baseline that is often missing in prior works. The second model is our simple adaptation of the Transformer architecture for tabular data, which outperforms other solutions on most tasks. Both models are compared to many existing architectures on a diverse set of tasks under the same training and tuning protocols. We also compare the best DL models with Gradient Boosted Decision Trees and conclude that there is still no universally superior solution. The source code is available at <https://github.com/yandex-research/rtdl>.

<https://arxiv.org/abs/2106.11959>

Table 1: Dataset properties. Notation: “RMSE” ~ root-mean-square error, “Acc.” ~ accuracy.

	CA	AD	HE	JA	HI	AL	EP	YE	CO	YA	MI
#objects	20640	48842	65196	83733	98050	108000	500000	515345	581012	709877	1200192
#num. features	8	6	27	54	28	128	2000	90	54	699	136
#cat. features	0	8	0	0	0	0	0	0	0	0	0
metric	RMSE	Acc.	Acc.	Acc.	Acc.	Acc.	Acc.	RMSE	Acc.	RMSE	RMSE
#classes	–	2	100	4	2	1000	2	–	7	–	–

Table 4: Results for ensembles of GBDT and the main DL models. For each model-dataset pair, the metric value averaged over three ensembles is reported. See supplementary for standard deviations. Notation follows Table 3.

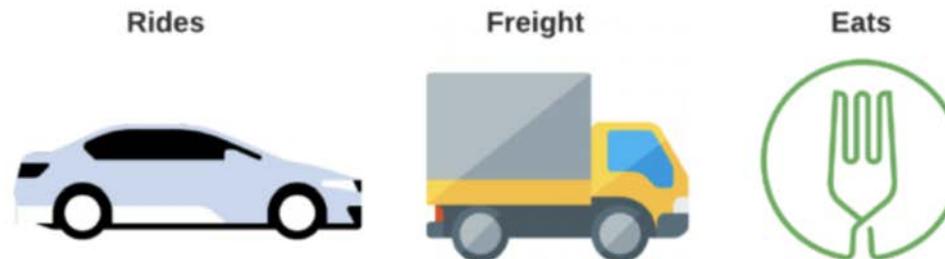
	CA ↓	AD ↑	HE ↑	JA ↑	HI ↑	AL ↑	EP ↑	YE ↓	CO ↑	YA ↓	MI ↓
Default hyperparameters											
XGBoost	0.462	0.874	0.348	0.711	0.717	0.924	0.8799	9.192	0.964	0.761	0.751
CatBoost	0.428	0.873	0.386	0.724	0.728	0.948	0.8893	8.885	0.910	0.749	0.744
FT-Transformer	0.454	0.860	0.395	0.734	0.731	0.966	0.8969	8.727	0.973	0.747	0.742
Tuned hyperparameters											
XGBoost	0.431	0.872	0.377	0.724	0.728	–	0.8861	8.819	0.969	0.732	0.742
CatBoost	0.423	0.874	0.388	0.727	0.729	–	0.8898	8.837	0.968	0.740	0.741
ResNet	0.478	0.857	0.398	0.734	0.731	0.966	0.8976	8.770	0.967	0.751	0.745
FT-Transformer	0.448	0.860	0.398	0.739	0.731	0.967	0.8984	8.751	0.973	0.747	0.743

DeepETA: How Uber Predicts Arrival Times Using Deep Learning

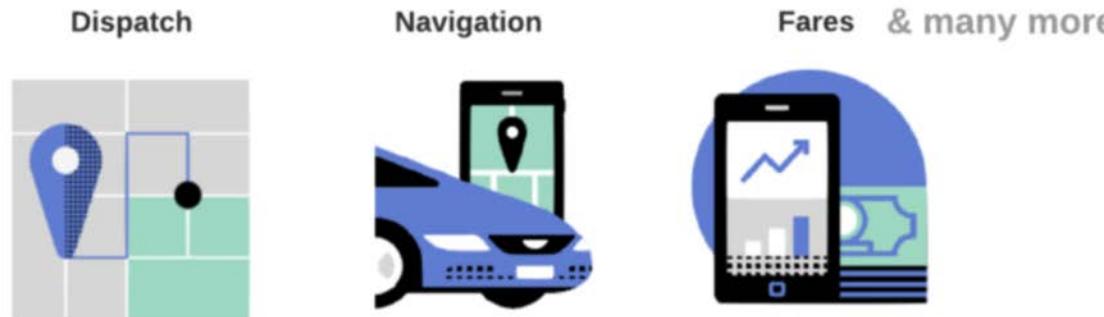
Xinyu Hu, Olcay Cirit, Tanmay Binaykiya, and Ramit Hora

0

February 10, 2022



ETAs power **Uber** experiences



<https://eng.uber.com/deepeta-how-uber-predicts-arrival-times/>

For several years, Uber used gradient-boosted decision tree ensembles to refine ETA predictions. The ETA model and its training dataset grew steadily larger with each release. To keep pace with this growth, Uber's [Apache Spark™](#) team contributed upstream improvements [1, 2] to XGBoost to allow the model to grow ever deeper, making it one of the largest and deepest XGBoost ensembles in the world at that time. Eventually, we reached a point where increasing the dataset and model size using XGBoost became untenable. To continue scaling the model and improving accuracy, we decided to explore deep learning because of the relative ease of scaling to large datasets using [data-parallel SGD](#) [3]. To justify switching to deep learning we needed to overcome three main challenges:

- **Latency:** The model must return an ETA within a few milliseconds at most.
- **Accuracy:** The mean absolute error (MAE) must improve significantly over the incumbent XGBoost model.
- **Generality:** The model must provide ETA predictions globally across all of Uber's lines of business such as mobility and delivery.

Cautionary notes

- “There is no free lunch.”
- The “**No Free Lunch**” theorem states that there is no one model that works best for every problem.
- Depends on
 - Nature of the task
 - Nature of the data
 - Amount of data
- Which model is the best?
 - Try it on your problem.

“Deep learning is not magical.”

What can deep learning do well?

- Unstructured data
 - Image, text (certain tasks), sensor signals, satellite image
- Generative tasks is pretty much only doable via deep learning

That's not so helpful...

Tech Support

...RESTART MY COMPUTER?
I KNOW YOU HAVE A SCRIPT
TO FOLLOW, BUT THE UPLINK
LIGHT ON THE MODEM IS GOING
OFF EVERY FEW HOURS. THE
PROBLEM IS BETWEEN YOUR
OFFICE AND THE MODEM.



MY COMPUTER HAS NOTHING
TO DO WITH ... OK, WHATEVER,
I "RESTARTED MY COMPUTER".
IT'S STILL DOWN, AND EVEN
IF IT COMES BACK, IT'S
GOING TO DIE AGAIN IN A
FEW HOURS, BECAUSE YOUR-

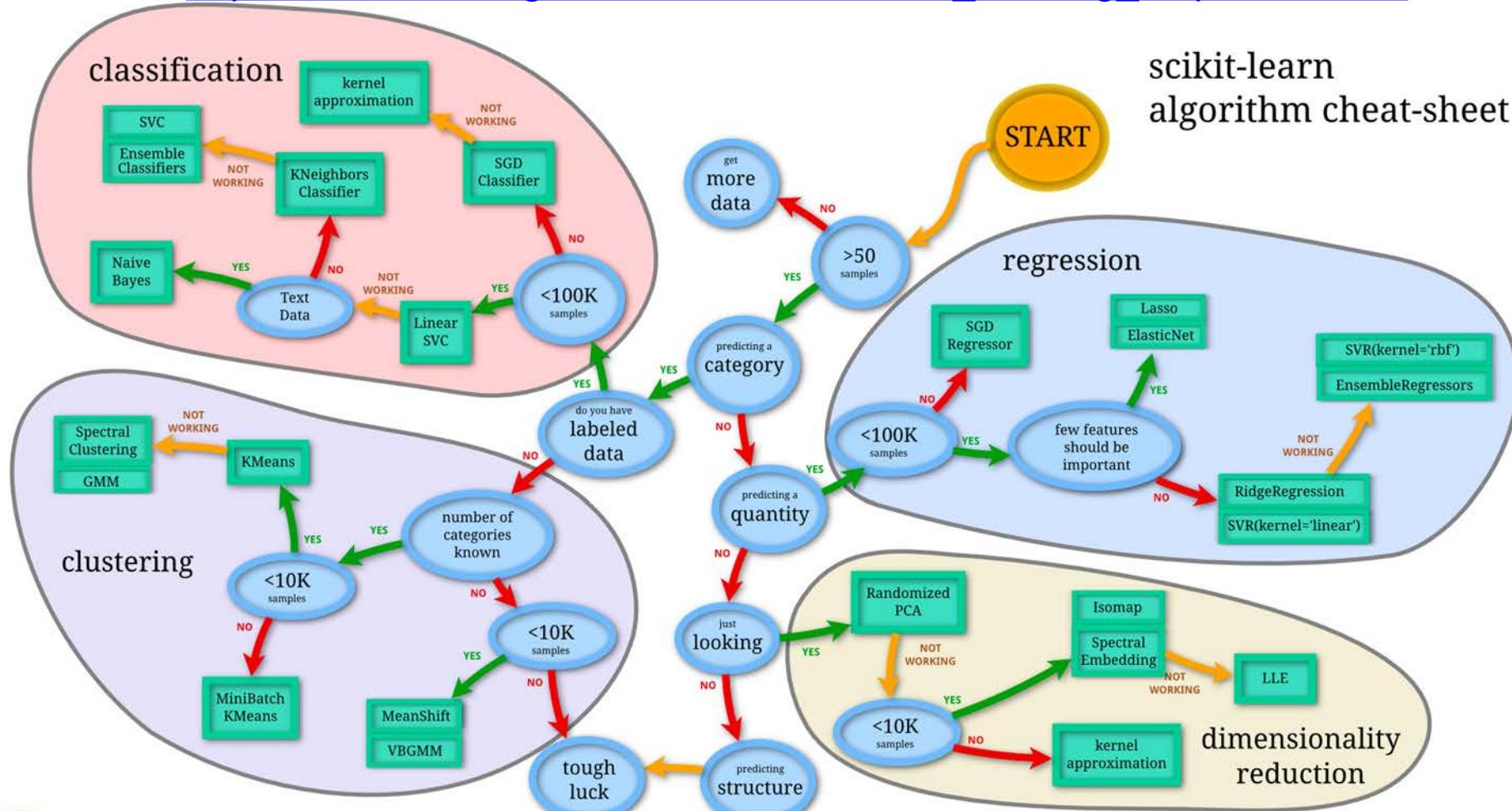


I DON'T HAVE A START MENU.
THIS IS A HAIKU INSTALL,
BUT THAT'S NOT IMPORT-
I
HAIKU? IT'S AN EXPERIMENTAL
OS THAT I ... OH, NEVER MIND.



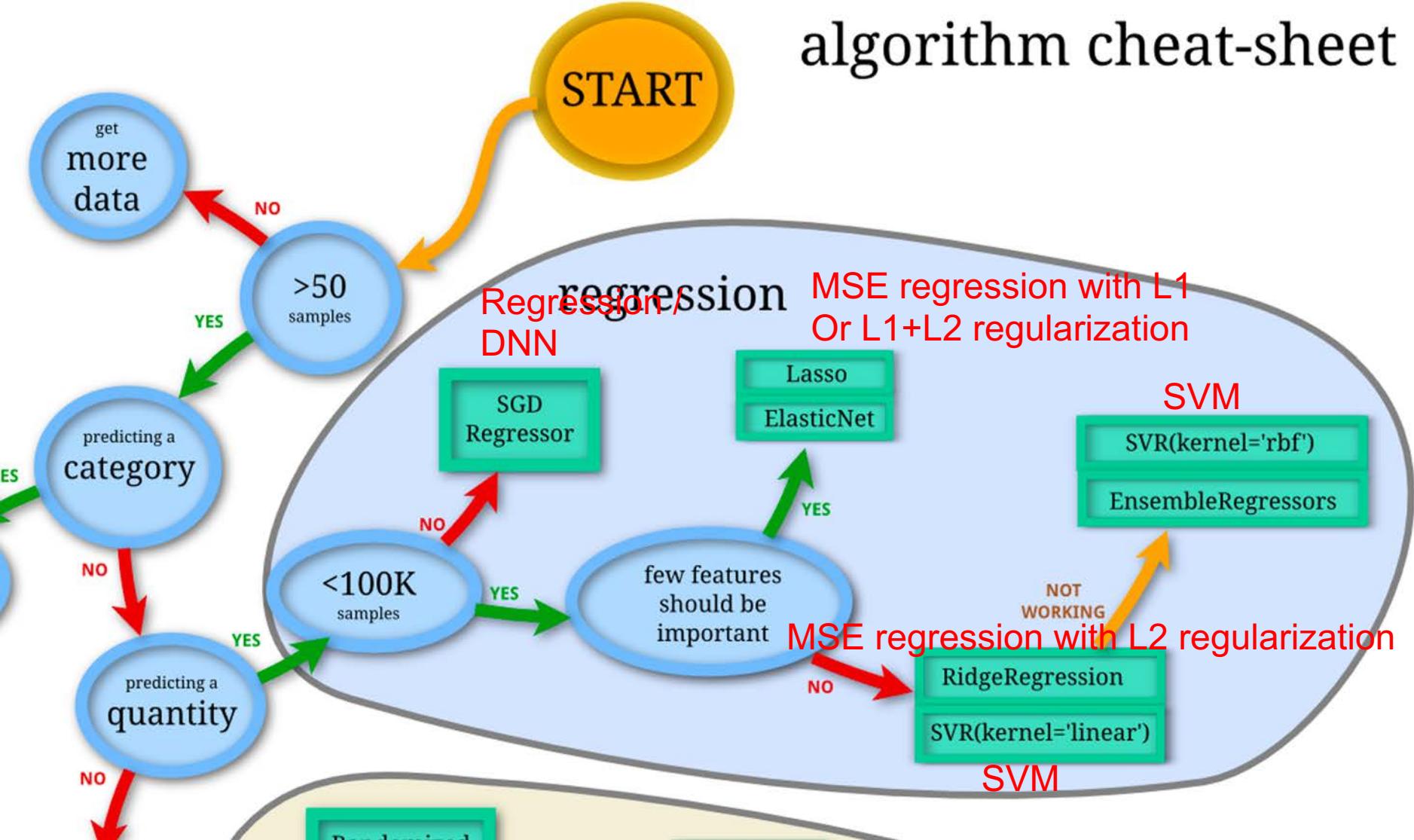
<https://xkcd.com/806/>

scikit-learn algorithm cheat-sheet



Note: treat 100k, 10k samples as a guideline.
 These numbers can go bigger or smaller depending on feature dimension and number of classes.

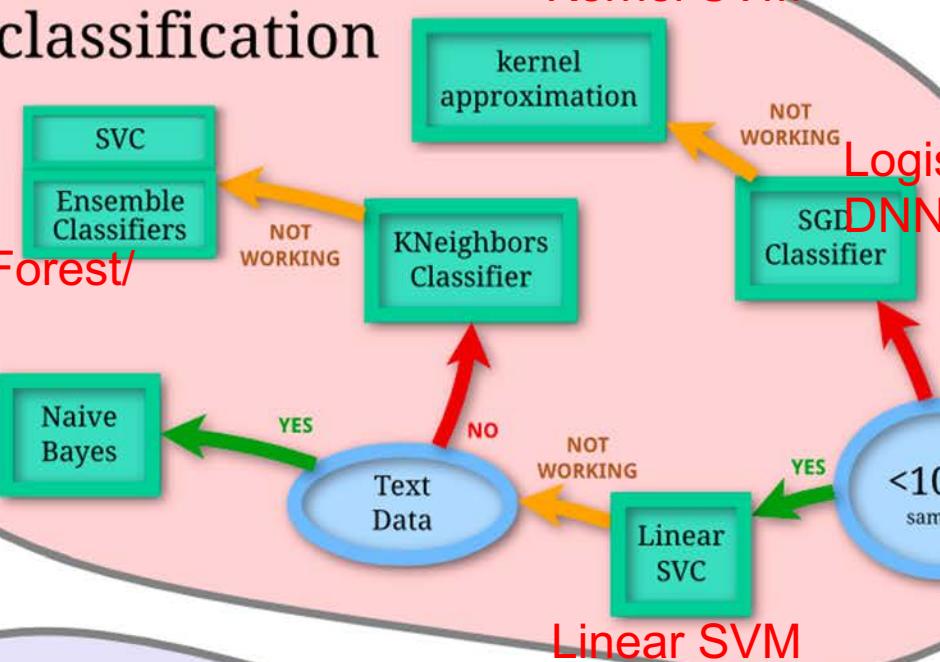
scikit-learn algorithm cheat-sheet



classification

Random Forest/
XGBoost

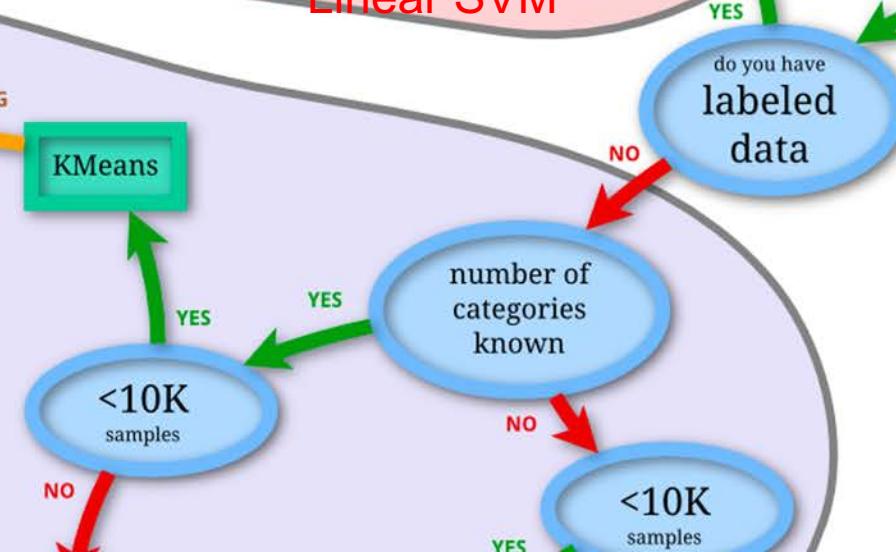
Kernel SVM



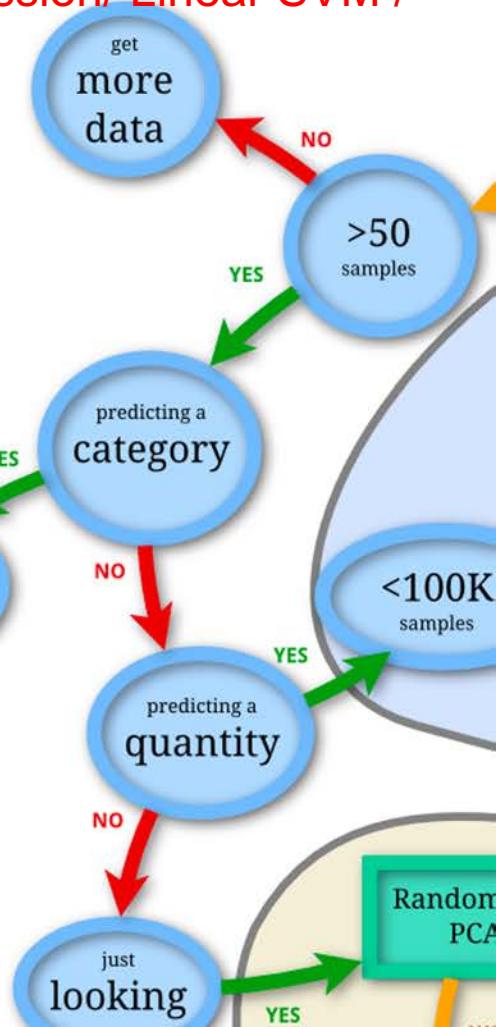
Logistic regression/ Linear SVM /
DNN

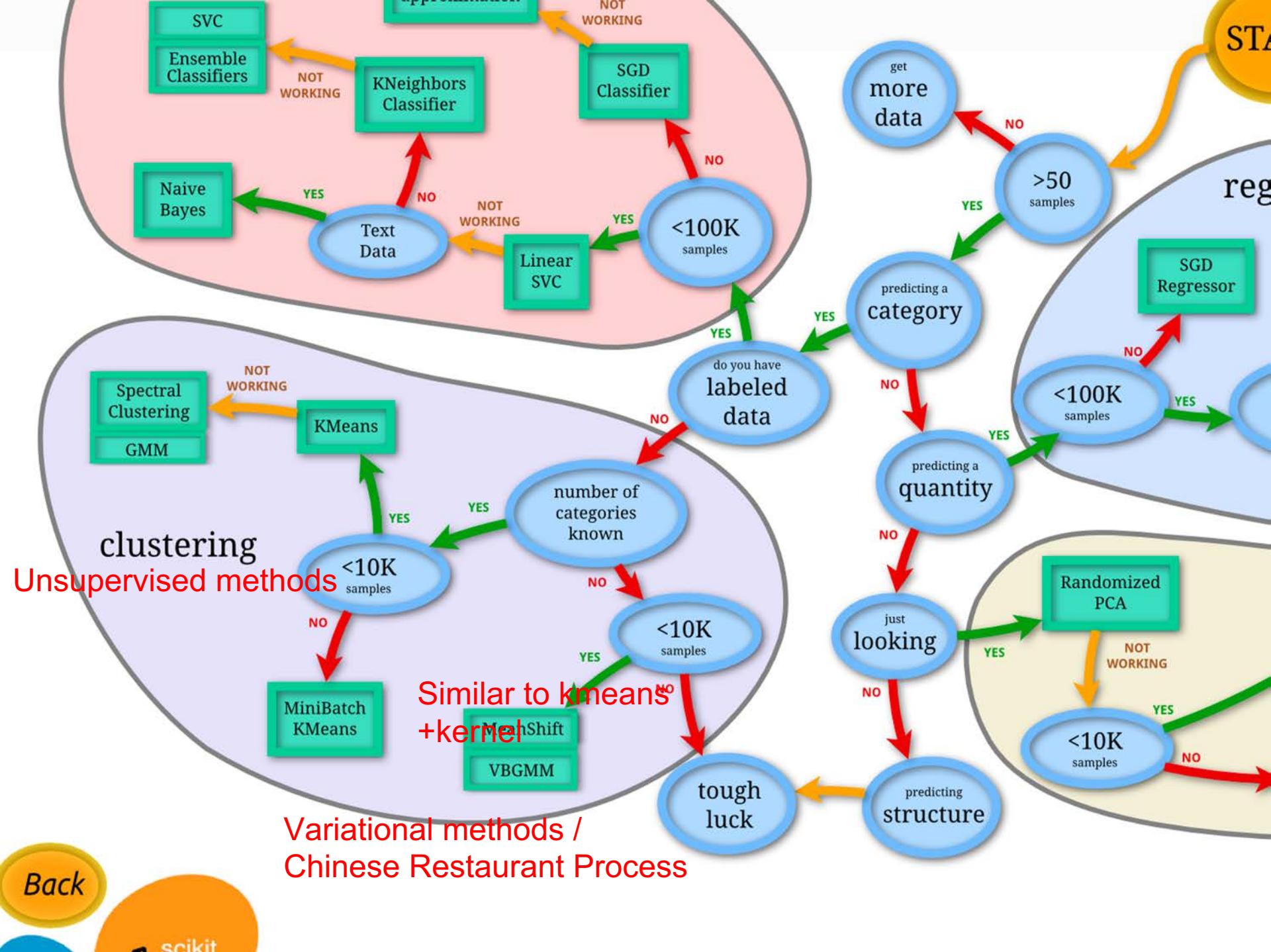
clustering

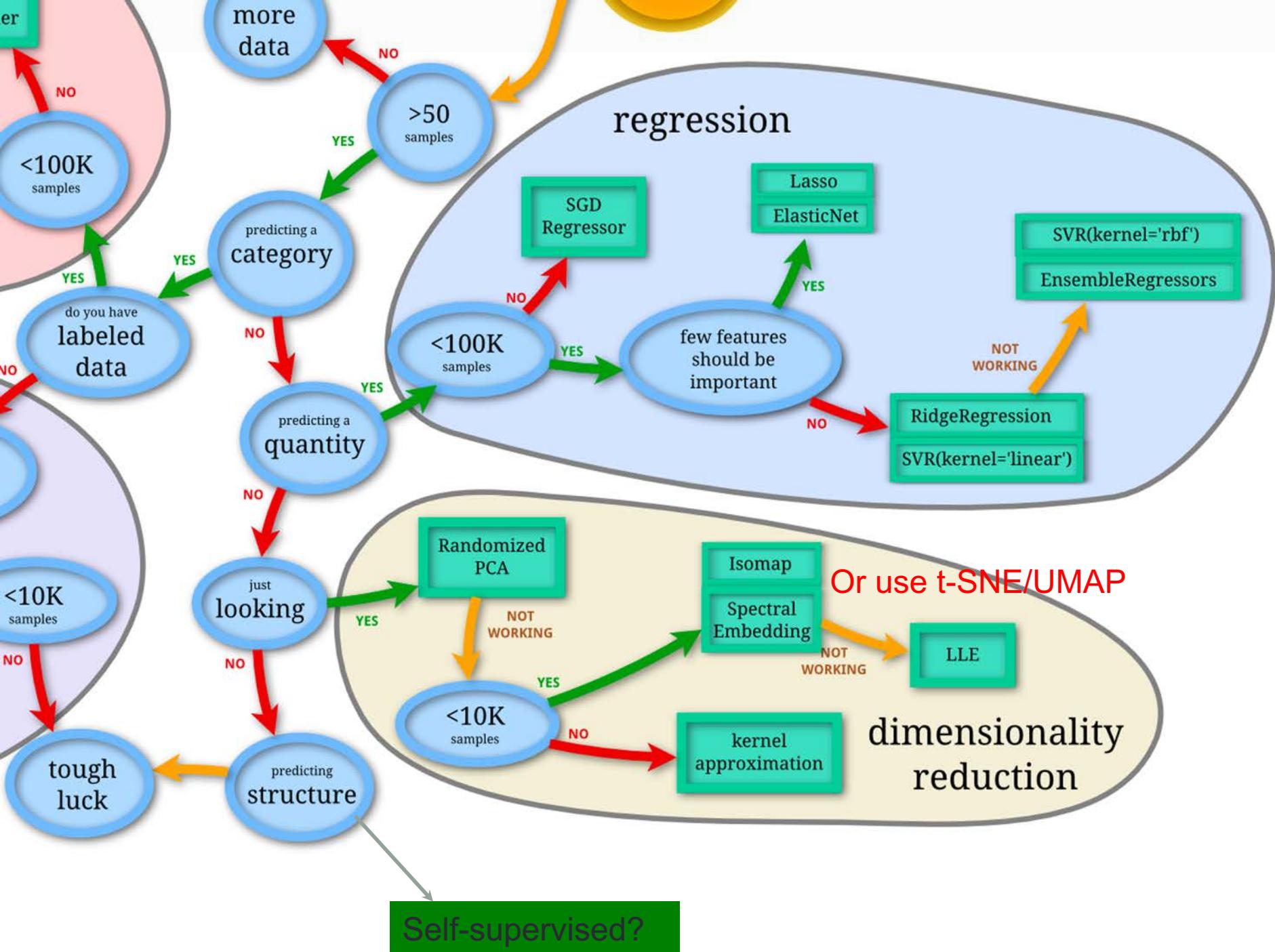
Spectral
Clustering
GMM



Random
PCA







Do your literature review

- Check for the closest task
 - Does not need to be in the same domain
 - Ex: Strings of DNA -> NLP!
 - Most advances in many fields came from cross-fertilization
- Reading checklist
 - Amount of data
 - Number of classes
 - Data/classes
 - Features
 - Models
- Look at multiple papers

Literature review tricks

- Search forward and backward in

<https://scholar.google.com/>

- Citations
- Cited by

[BOOK] **Neural network design**

HB Demuth, MH Beale, O De Jess, MT Hagan - 2014 - dl.acm.org

Abstract This book, by the authors of the **Neural Network** Toolbox for MATLAB, provides a clear and detailed coverage of fundamental **neural network** architectures and learning rules.

In it, the authors emphasize a coherent presentation of the principal **neural** networks,

☆ 99 [Cited by 7914](#) Related articles All 3 versions ➞

Reading between the lines

- If you don't understand something, re-read
 - Average of 5+ times to understand a paper completely
- Print it out, keep a pen/pencil at hand, and break down equations
- Try to explain it with what you already know

Elastic net Loss function $\min_w \frac{1}{2n_{samples}} \|Xw - y\|_2^2 + \alpha\rho\|w\|_1 + \frac{\alpha(1-\rho)}{2}\|w\|_2^2$

Reading between the lines

- If you don't understand something, re-read
 - Average of 5+ times to understand a paper completely
- Print it out, keep a pen/pencil at hand, and break down equations
- Try to explain it with what you already know

Elastic net Loss function $\min_w \frac{1}{2n_{samples}} \|Xw - y\|_2^2 + \alpha\rho\|w\|_1 + \frac{\alpha(1-\rho)}{2}\|w\|_2^2$

MSE Loss

L1

L2

Rule #0: maybe no ML?

- Sometimes you don't need a machine learning model
 - Launch a simpler system to collect data
 - If possible, have the user label your data.

mail.google.com says

It seems like you forgot to attach a file.

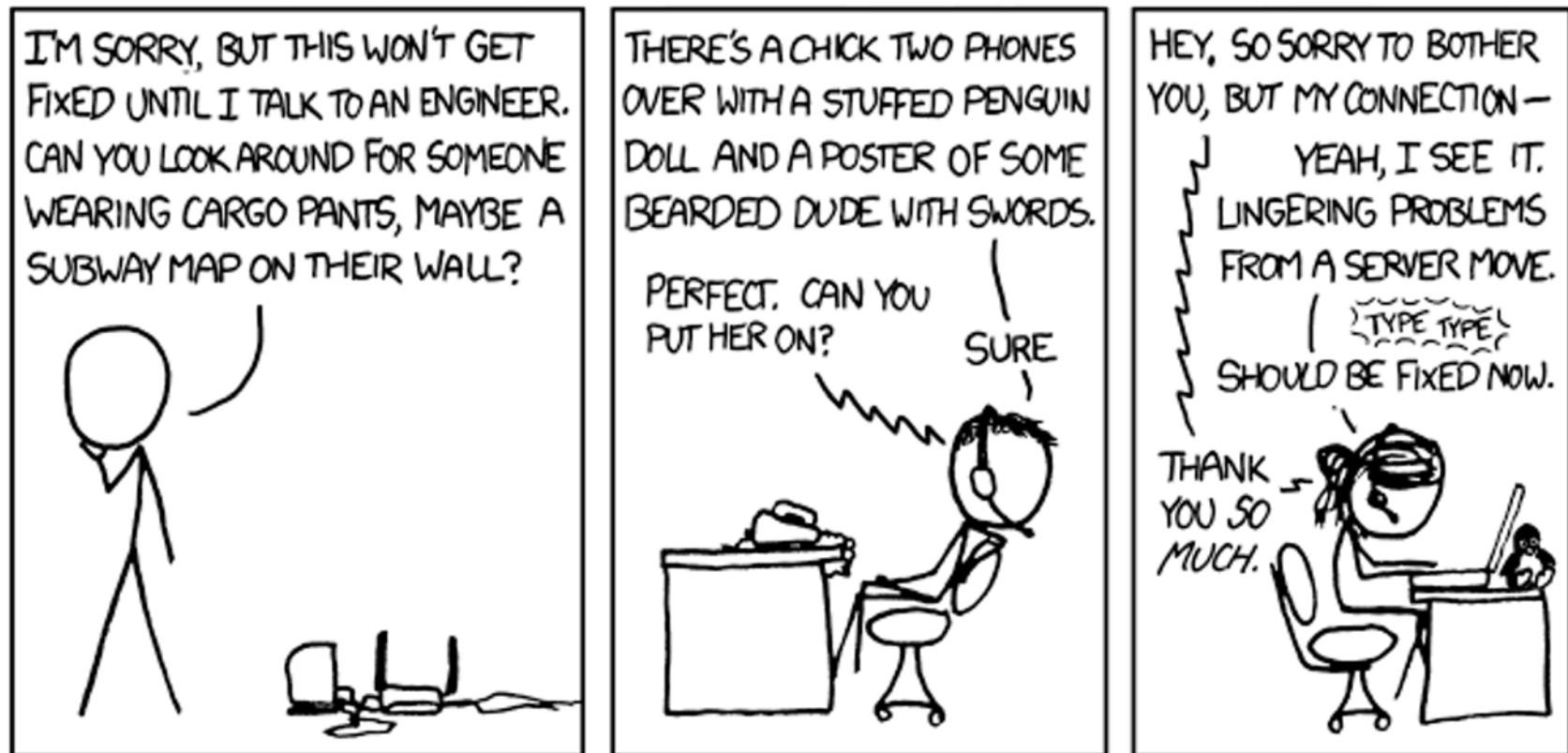
You wrote "find attached" in your message, but there are no files attached. Send anyway?

Cancel

OK

How to improve my results?

- More tech support

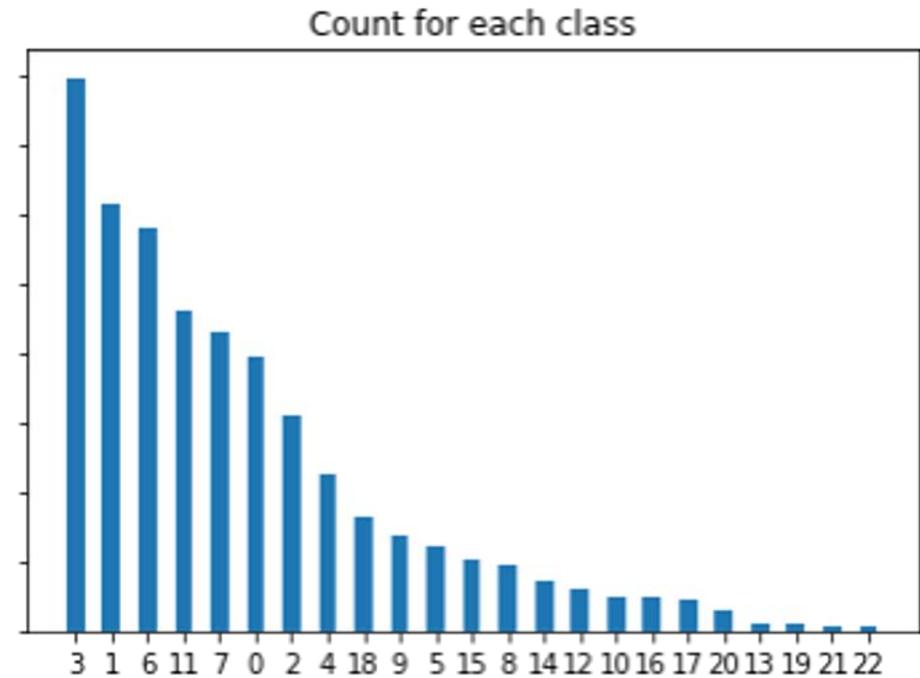


Diagnosis and prescription

- Is my task proper?
- Should I add/remove feature XXX?
- Is my loss function appropriate?
- Is my model overfitting/underfitting?
- Should I try XXX?

Is my task proper? Do you suffer from class imbalance?

- Throwing away
- Refactoring
 - Split
 - Merge
- Data augmentation
- Biasing
 - Weighting the loss function
 - Bias in mini-batch sampling



Diagnosis

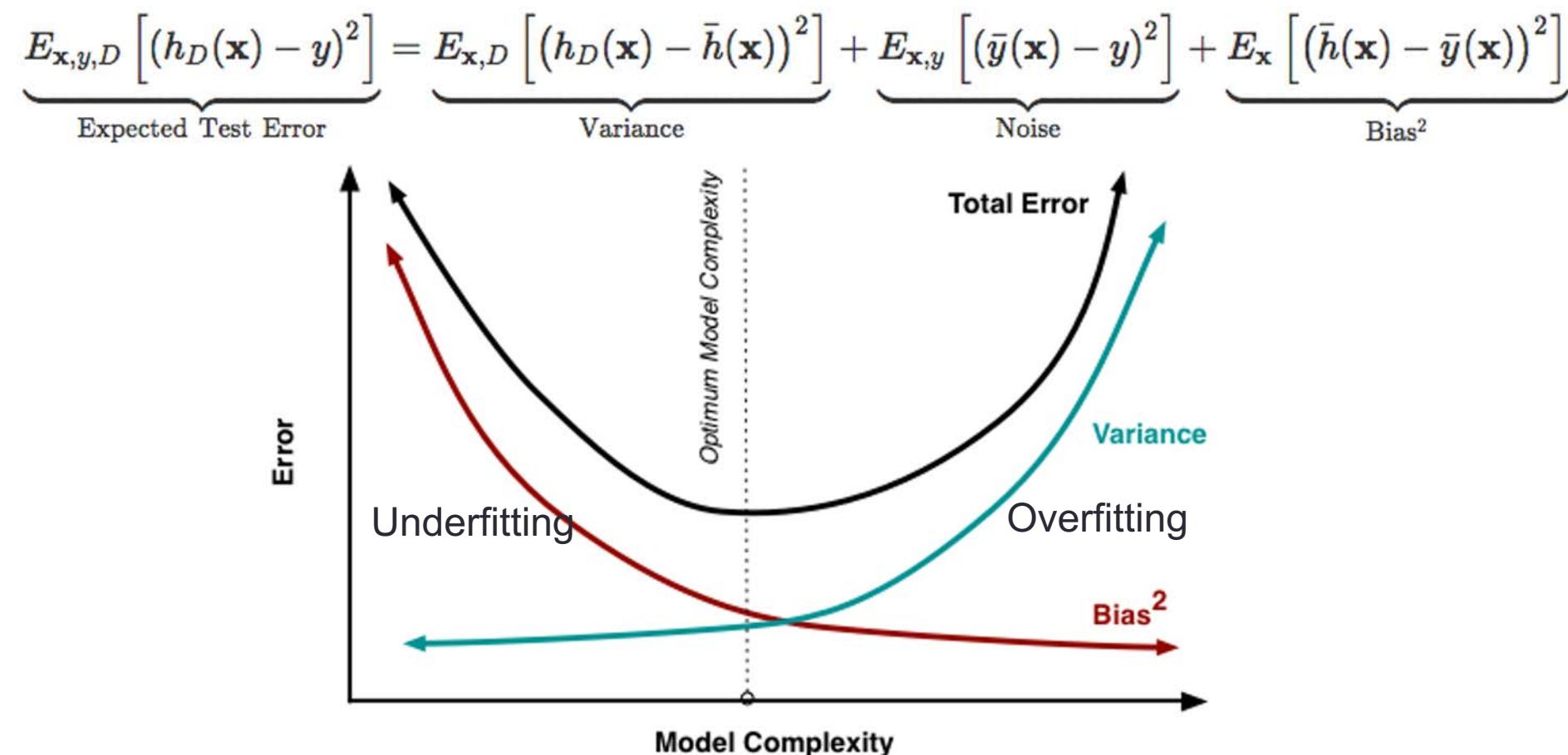
- Is my task proper?
- Should I add/remove feature XXX?
- Is my loss function appropriate?
- Is my model overfitting/underfitting?
- Should I try XXX?

Short answer: Cross Validation

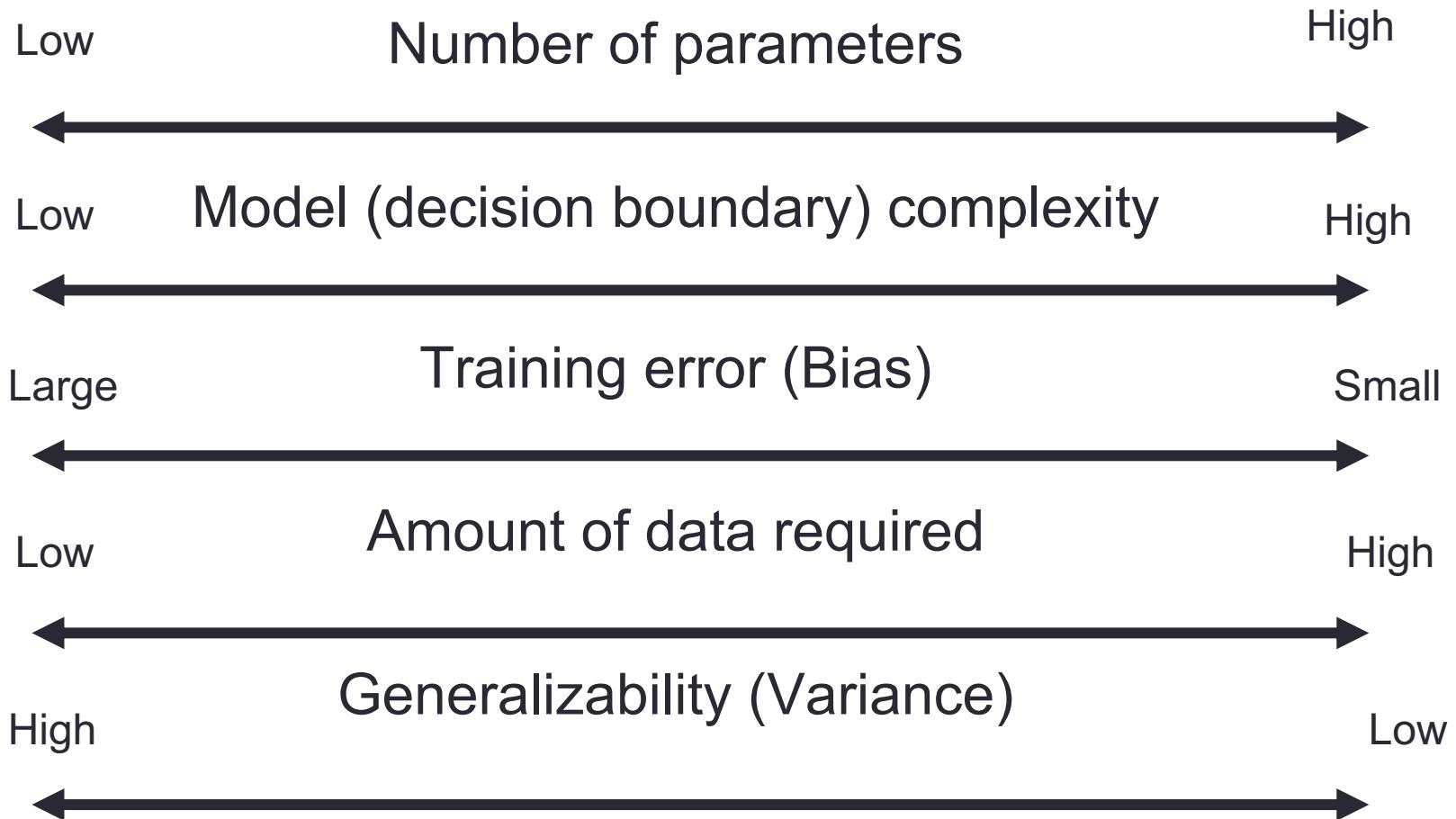
Can we do better?
Find the problem and fix it.

Understanding the bias variance trade-off

- Bias variance analysis can be helpful for diagnosis



Bias-Variance overview

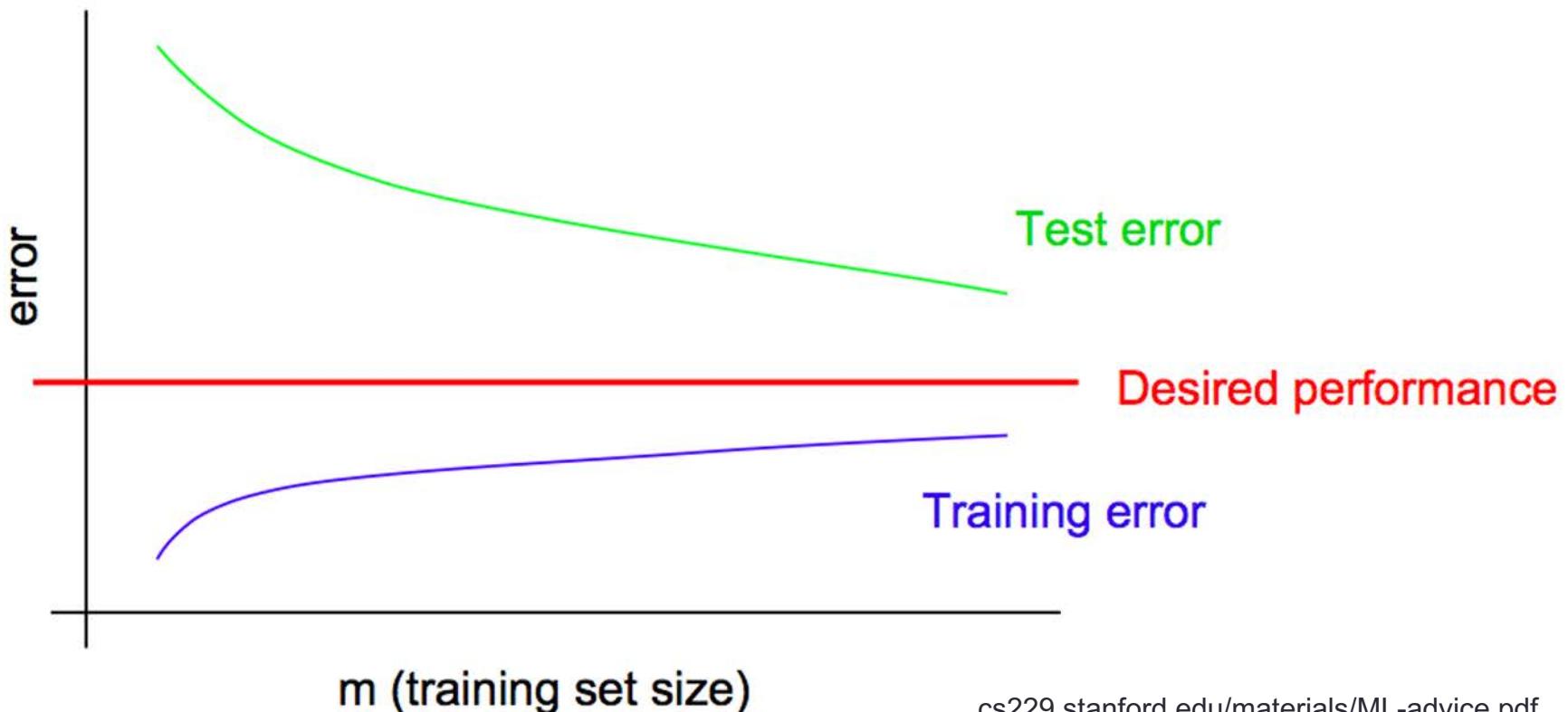


Bias-variance diagnosis

- Suppose the problem is either
 - Overfitting (high variance)
 - Too few features to classify properly (high bias)
- Symptoms
 - Variance: Training error is much lower than test/dev error.
 - Bias: Training error is also high

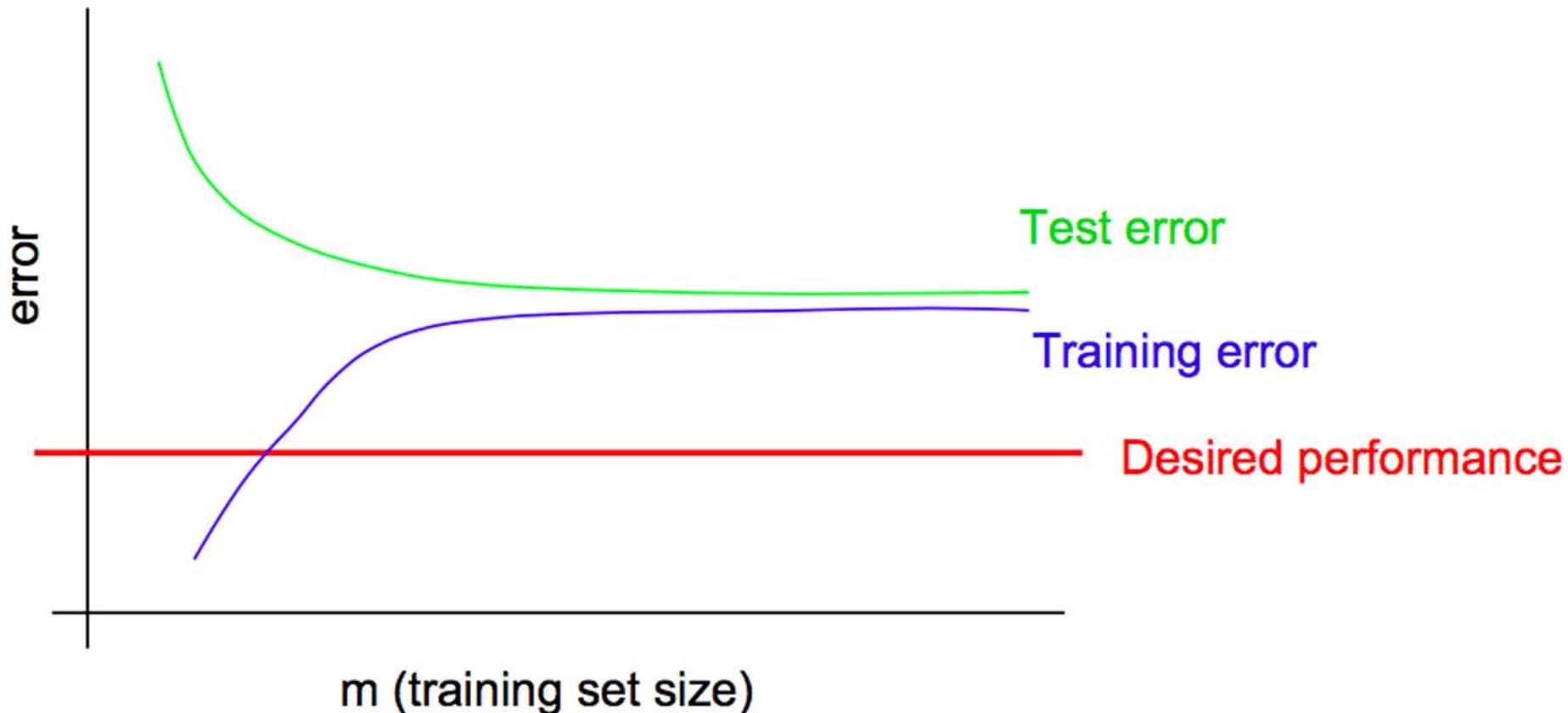
High variance case

- Solution:
 - Reduce overfitting: regularization, reduce features, etc
 - Get more training set



High bias case

- Solution: more features, more layers, bigger models



Desired performance?

- The goal of the application
 - Possibilities
 - Target performance to launch the product
 - Human performance on the task
 - Have A and B do the same task, measure the difference.
- Knowing human performance gives several advantages
 - Knows when to stop
 - Beating human is sometimes a goal, but
 - Some errors might be labeling errors or judgment calls



Error diagnosis

- You're making a cat classifier. (cat/not cat)
- It sucks.
- You heard of this super new hype algorithm (for example: capsule network). Should you spend months to try it out?



This is not a cat



Looking at the errors

- Spend an hour or two looking at your errors. Identify why. Keep a table.

	Blurred	Weird angle	Notes
Pic1	x		Stuffed toy
Pic2	x		
Pic3	x		
Pic4		x	Top view
...
	68%	2%	

Solution: Use a method to sharpen the image. Train on blurry images.

The table categories can expand as you look through more pictures and see frequently occurring error cases. So keep notes.

Diagnosis summary

- Simple analysis of the data can help you notice underlying problems
- Bias-variance diagnosis is a general method that can be applied to most tasks
- Other diagnosis depends on the application and need some understanding of the algorithms
- Error analysis can help guide your model improvements

Taking the time to do diagnosis can help you save months of trying random things.

DATA SCIENCE - A CLASH OF TWO PARADIGMS

1. *The scientific paradigm*

- *What should the world be like before I can answer my research question?*

2. *The data-centric paradigm*

- *How best to fit the data so as to maximize success on the training set.*

Lecture given by: Judea Pearl

Questions beyond classification/regression

Can I trust my model?

Why does it give this answer?

ML products

Customer facing

Recommendation systems, maps (traffic estimate), speech2text

Best guess by the model
Mostly automatic (check deposit by app)



Internal facing

Loan applications, demand forecasting

Can say “I’m not sure”
Human in the loop. Machine-assisted

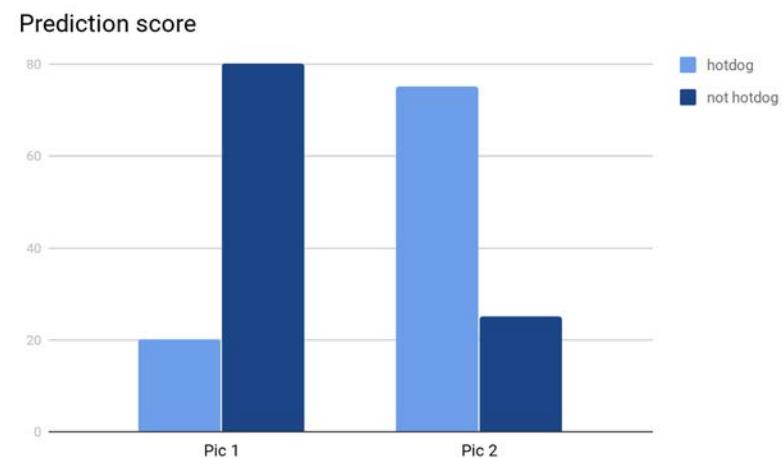
Requires confidence level



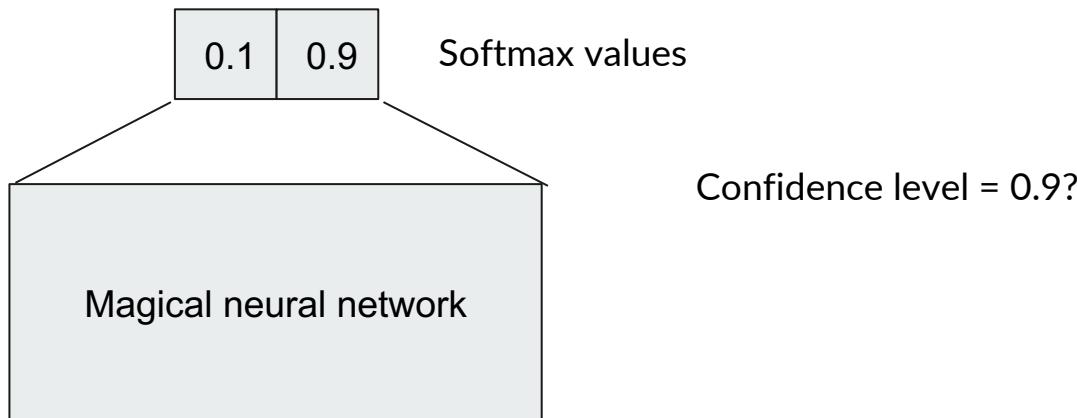
Confidence score

Practical models are not only accurate, but need to be able to state its confidence

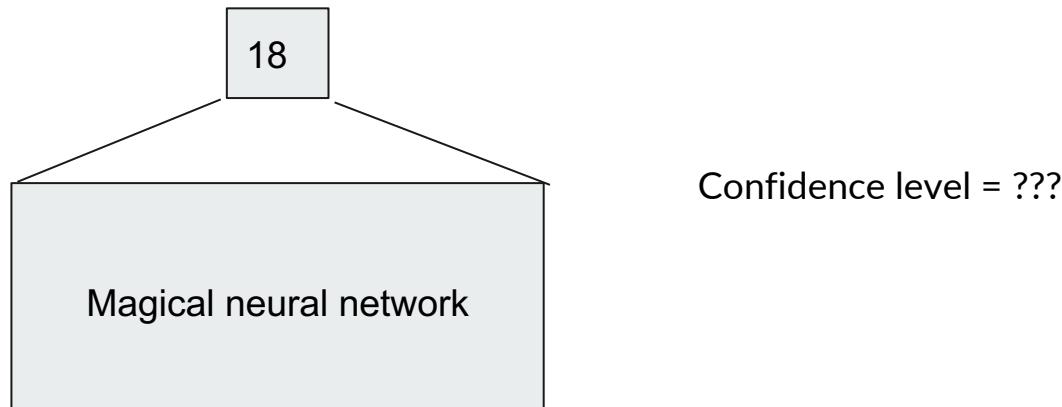
Confidence = probability of being correct



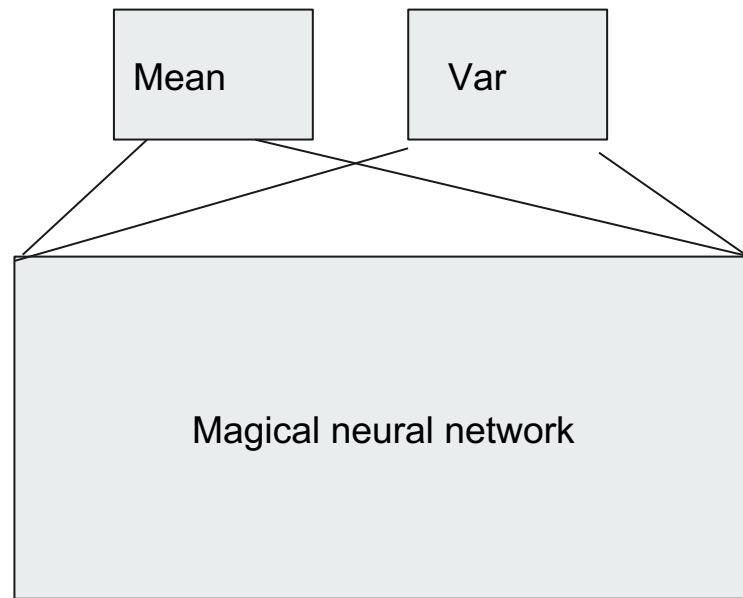
Naive way for confidence



What about regression?

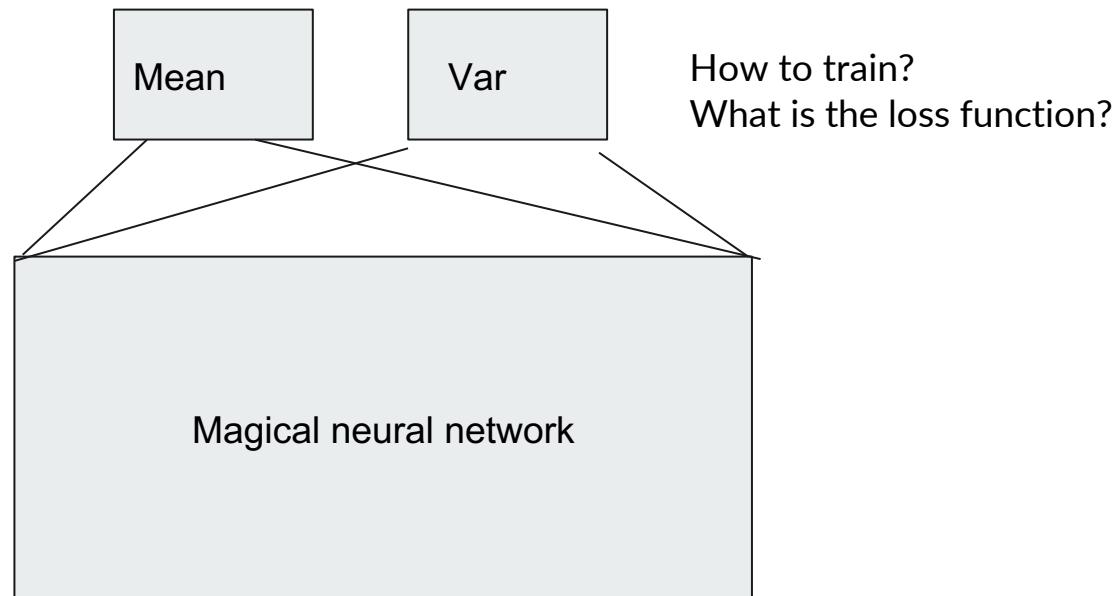


A Naive way for regression (1994!)

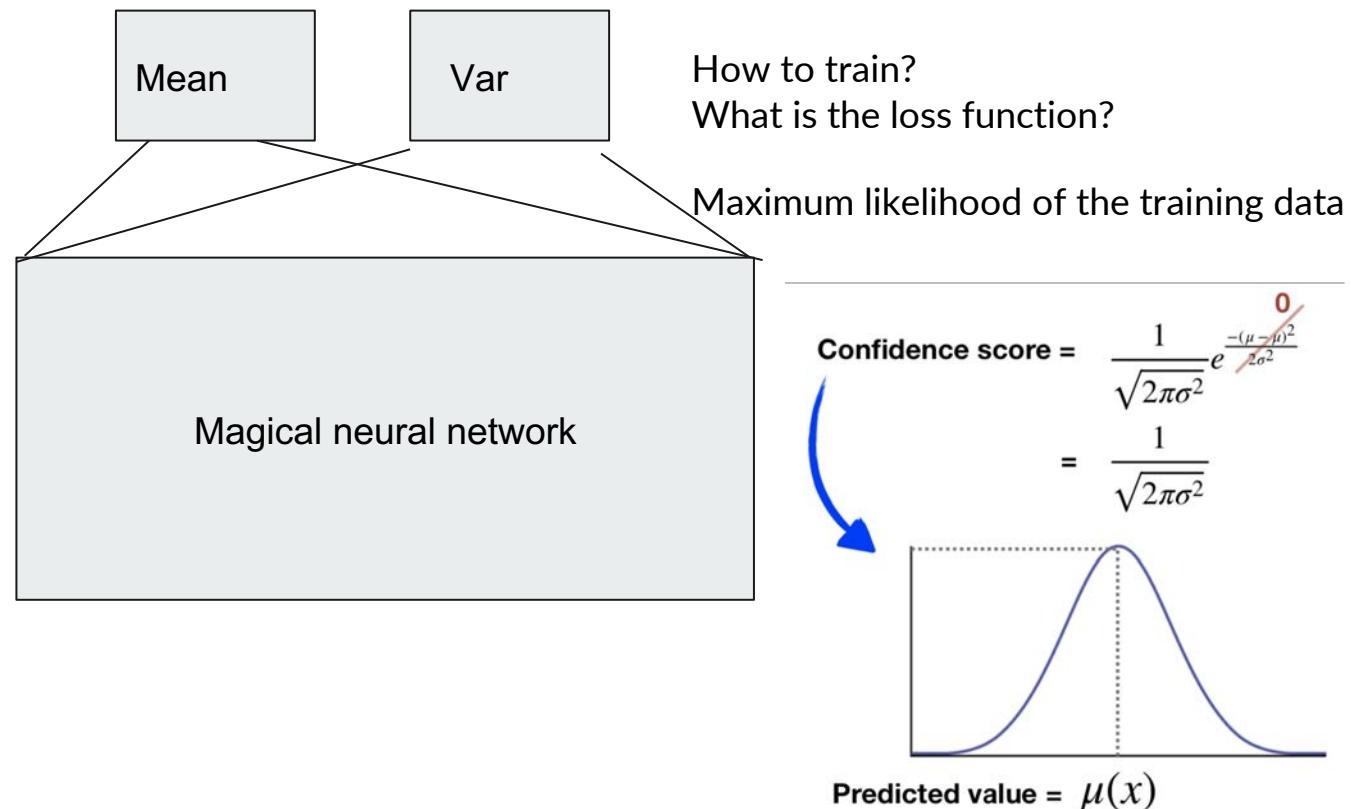


Estimating the mean and variance of the target probability distribution. IEEE 1994

A Naive way for regression (1994!)



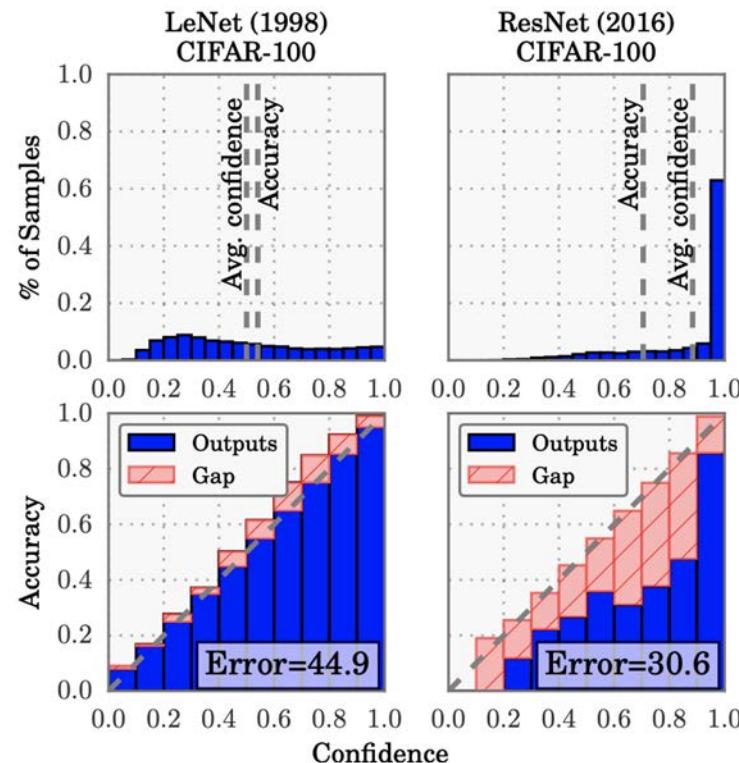
A Naive way for regression (1994!)



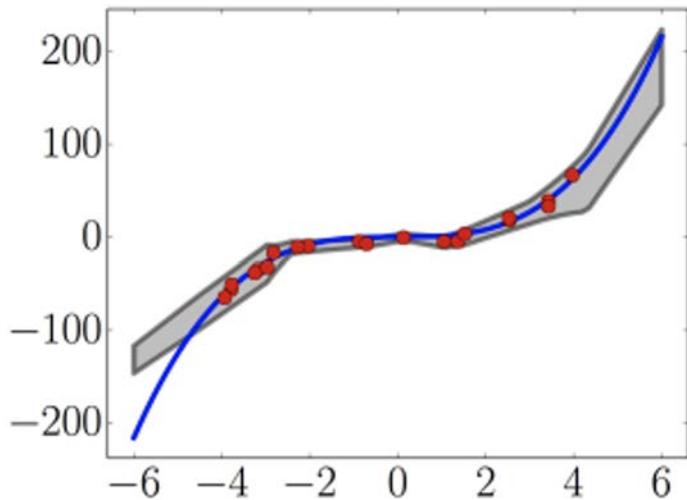
Estimating the mean and variance of the target probability distribution. IEEE 1994

Poorly calibrated confidence

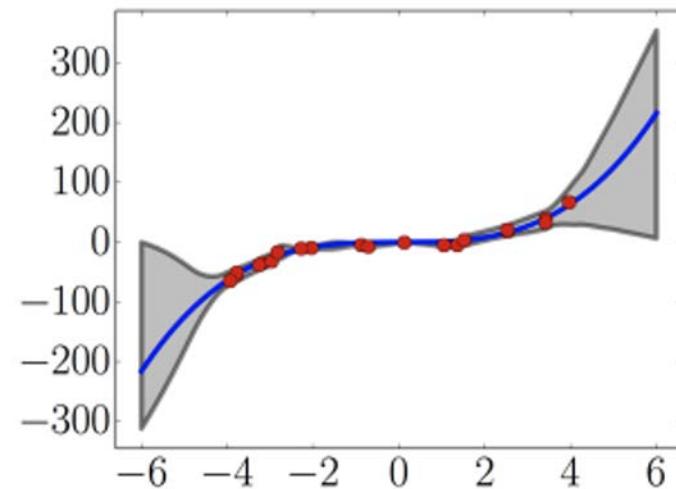
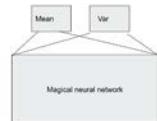
Deep Neural Networks are always overconfident!
Confidence = Probability of being correct



Out of distribution problem



output from predicting variance via
maximum likelihood



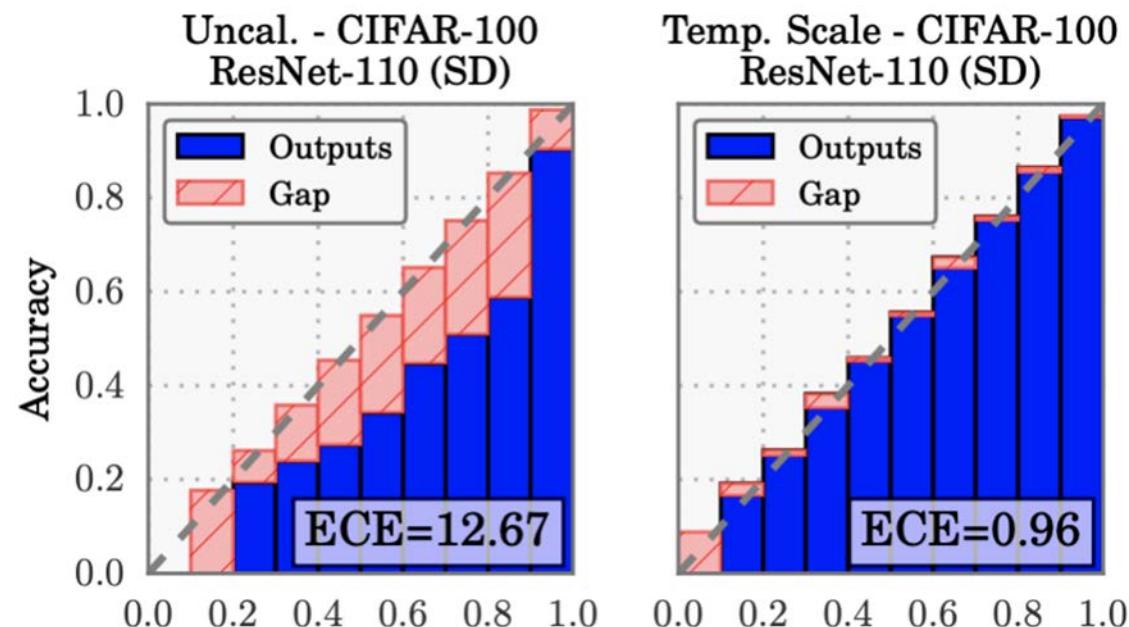
output from ensemble

Neural network calibration

Make the confidence output follows the probability of being correct.

How?

Need a separate training set to train the calibration (calibration set)



Overview of methods for calibration

1 Calibration

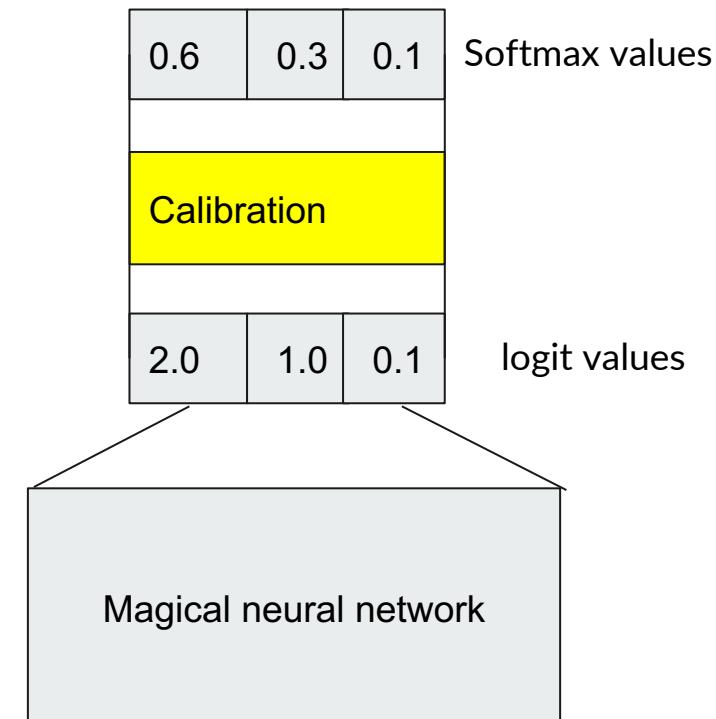
2 Ensemble

 2.1 Combining models

 2.2 Bayesian neural networks

Calibration

Post processing after getting pre-softmax output (logit)



Calibration - Temperature scaling

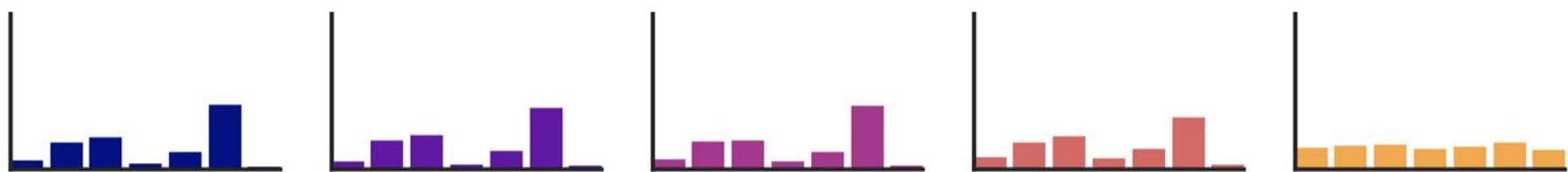
Add a temperature T to rescale the softmax

$$\hat{q}_i = \max_k \sigma_{\text{SM}}(\mathbf{z}_i/T)$$

T is tuned to maximize log likelihood on the calibration set

Low T

High T



Other calibration methods

Histogram binning

Bayesian binning into quartiles (BBQ)

Matrix and vector scaling (model on top of model)

Isotonic regression (model on top of model)

See also <https://scikit-learn.org/stable/modules/calibration.html>

Try different methods on your dataset. No absolute best.

Combining models

Create multiple models

Calculate mean and variance of the answers!

Multiple models can be just from different initializations

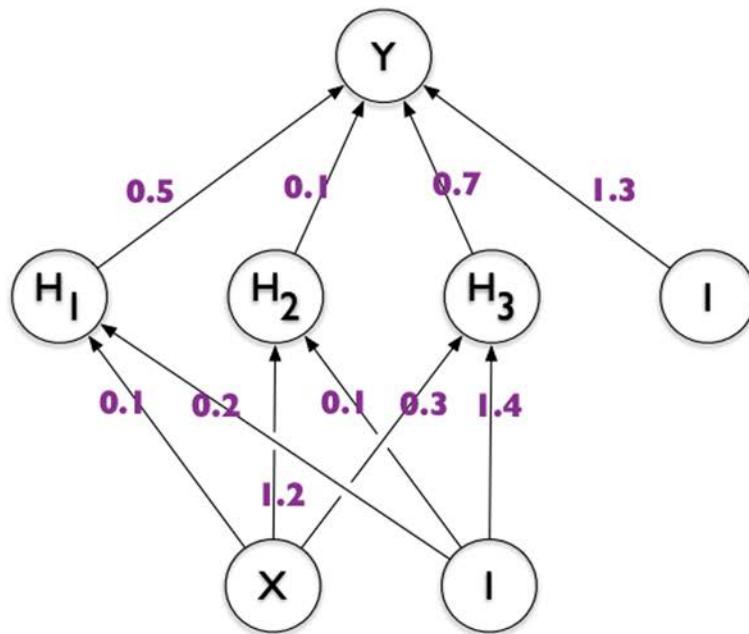
Cons: need to keep a lot of models around

How to get multiple models around?

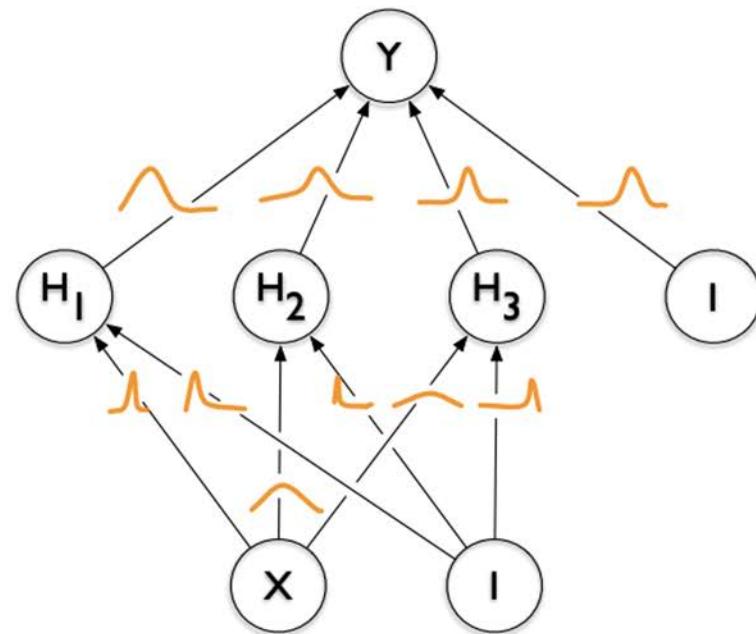
"Simple and scalable predictive uncertainty estimation using deep ensembles." 2017.

Bayesian Neural Network

Normal NN



Bayesian NN

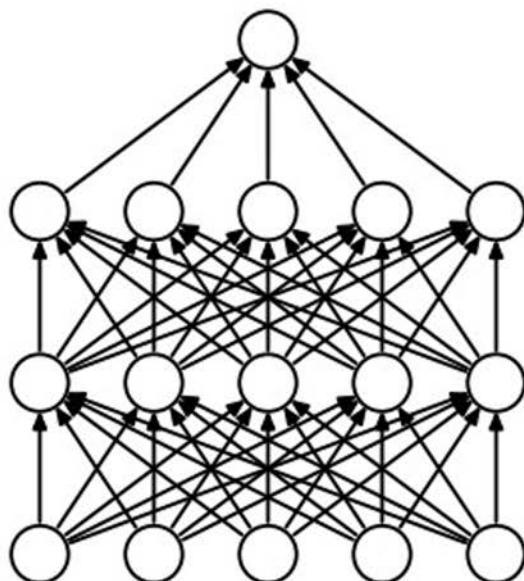


Problem: hard to do inference. Need to find $P(y|x, D) = \sum_{\theta} P(y|x, \theta) P(\theta|D)$

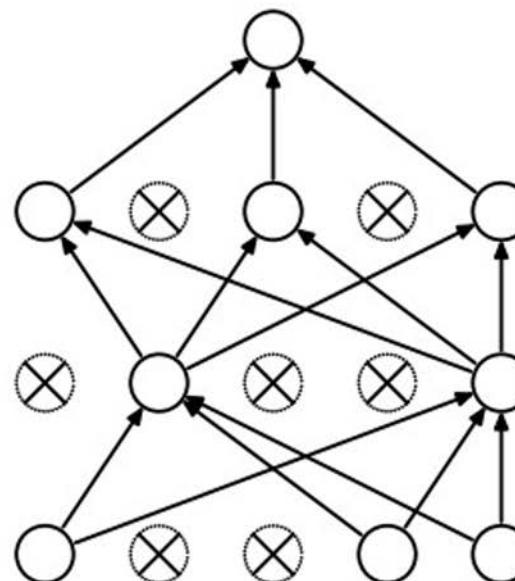
Monte Carlo dropout

Use dropout at test time to simulate different models

Sampling of dropout at inference time is the same as sampling weights from Bernoulli distribution



(a) Standard Neural Net



(b) After applying dropout.

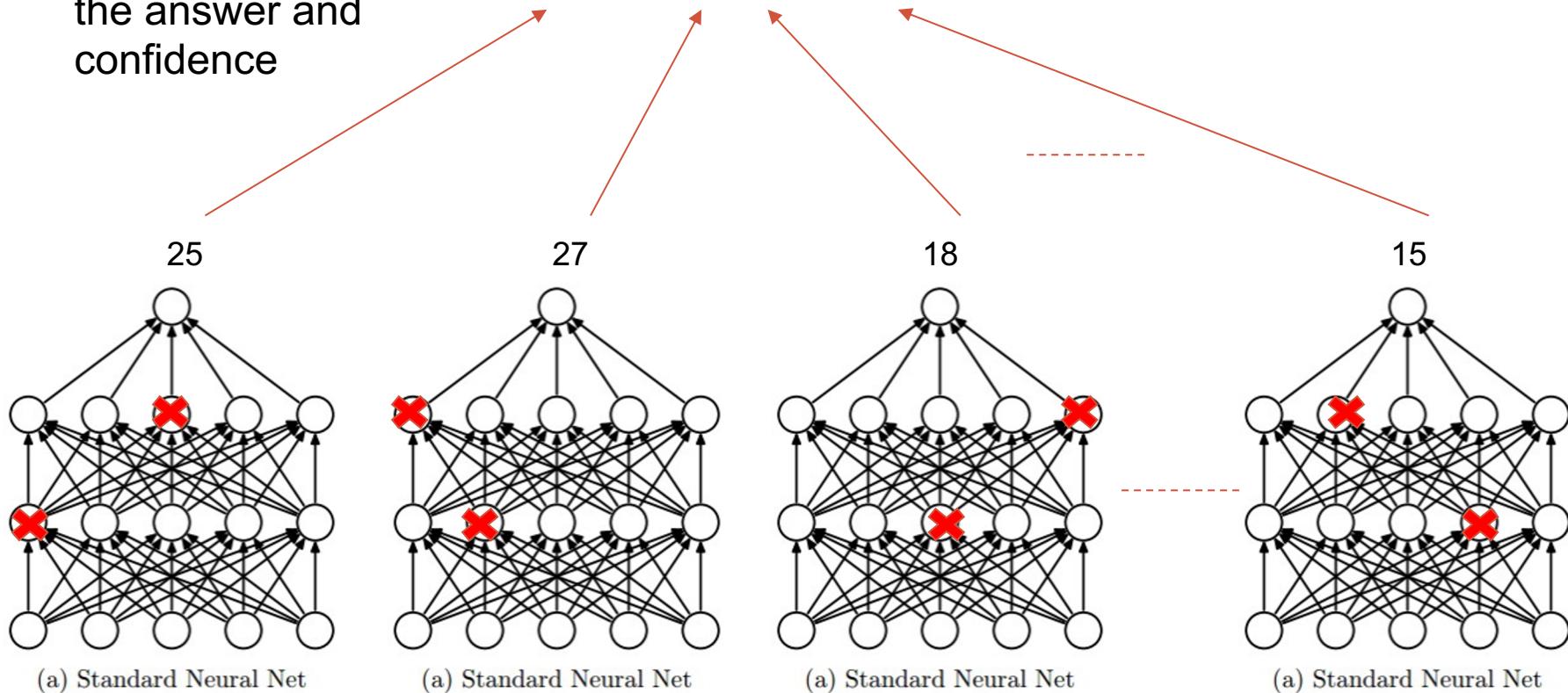
<https://www.depends-on-the-definition.com/model-uncertainty-in-deep-learning-with-monte-carlo-dropout/>

<https://arxiv.org/pdf/1506.02142.pdf> Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning

Monte Carlo dropout for confidence estimation

compute mean
and variance for
the answer and
confidence

20.2 ± 3.2



So we got the confidence, can we say why?



Two levels of understanding

Model level

Describes the model tendency

Talks about the behavior on training data

Output level

Attributes model decision for a given test sample to different features

Model level: Simple example

Logistic regression

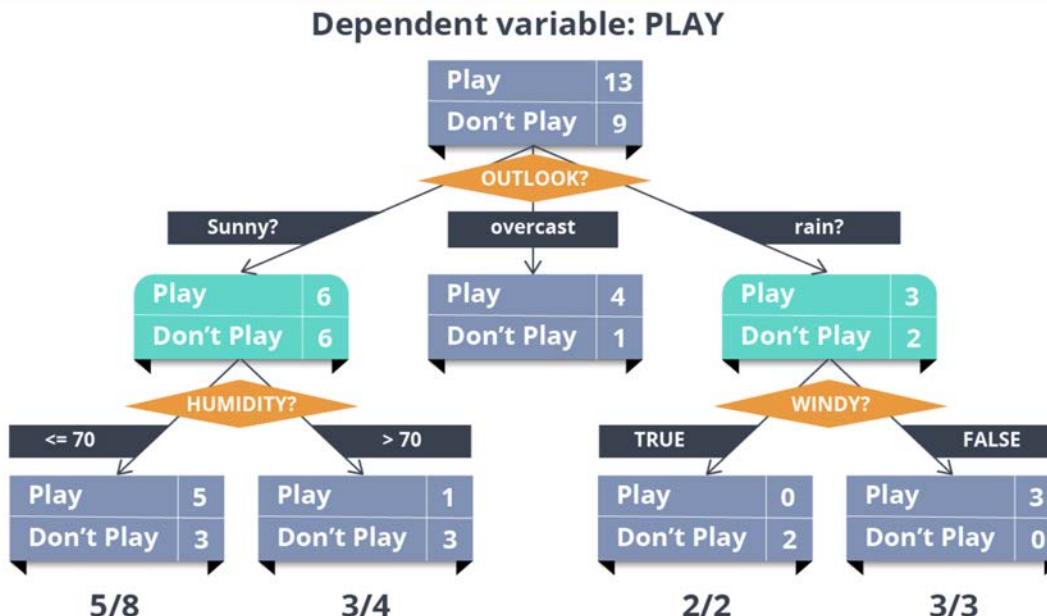
$$g(z') = \phi_0 + \sum_{i=1}^M \phi_i z'_i,$$

You can say which feature is important from size of the coefficients.

Pros: Simple, easy to understand

Cons: Only model linear effects

Model level: tree-based (feature importance in XGBoost)



Assign scores based on how the features are used

A node that splits better (better purity at children): high score

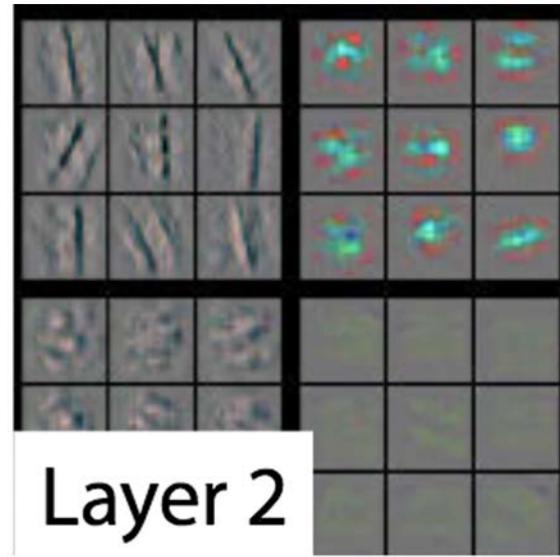
A node with more training samples: high weight

Model level: visualizing filters

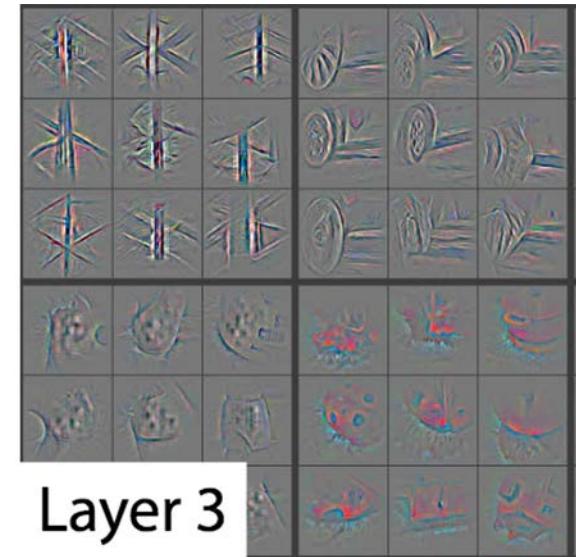
Plot out the weights of the CNN filters...not so useful in practice



Layer 1



Layer 2



Layer 3

Usefulness of model level

Feature selection

Gives you confidence that the model is learning reasonable things

Two levels of understanding

Model level

Describes the model tendency

Talks about the behavior on training data

Output level

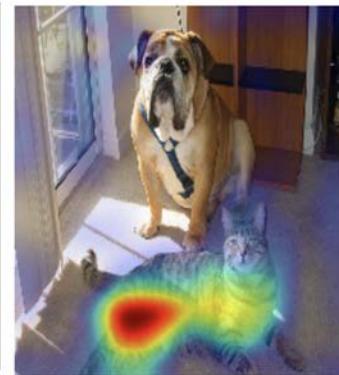
Attributes model decision for a given test sample to different features

Output level: key ideas

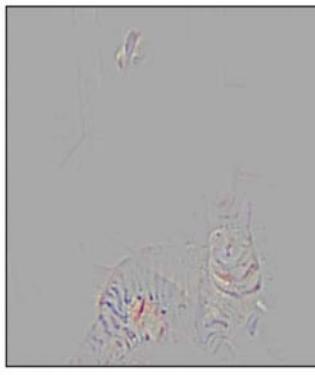
A feature is important if I tweak the feature and the output change a lot.

If I tweak the output, how does the gradient flows

Output level: Gradient-weighted Class Activation Mapping (Grad-CAM)



(c) Grad-CAM ‘Cat’



(d) Guided Grad-CAM ‘Cat’



(i) Grad-CAM ‘Dog’



(j) Guided Grad-CAM ‘Dog’



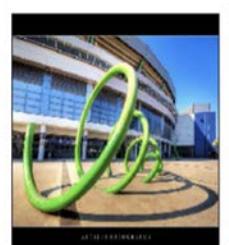
Ground truth: volcano



Ground truth: volcano



Ground truth: beaker



Ground truth: coil



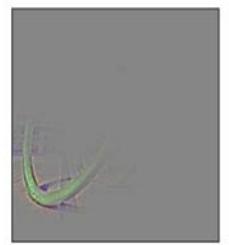
Predicted: sandbar
(a)



Predicted: car mirror
(b)

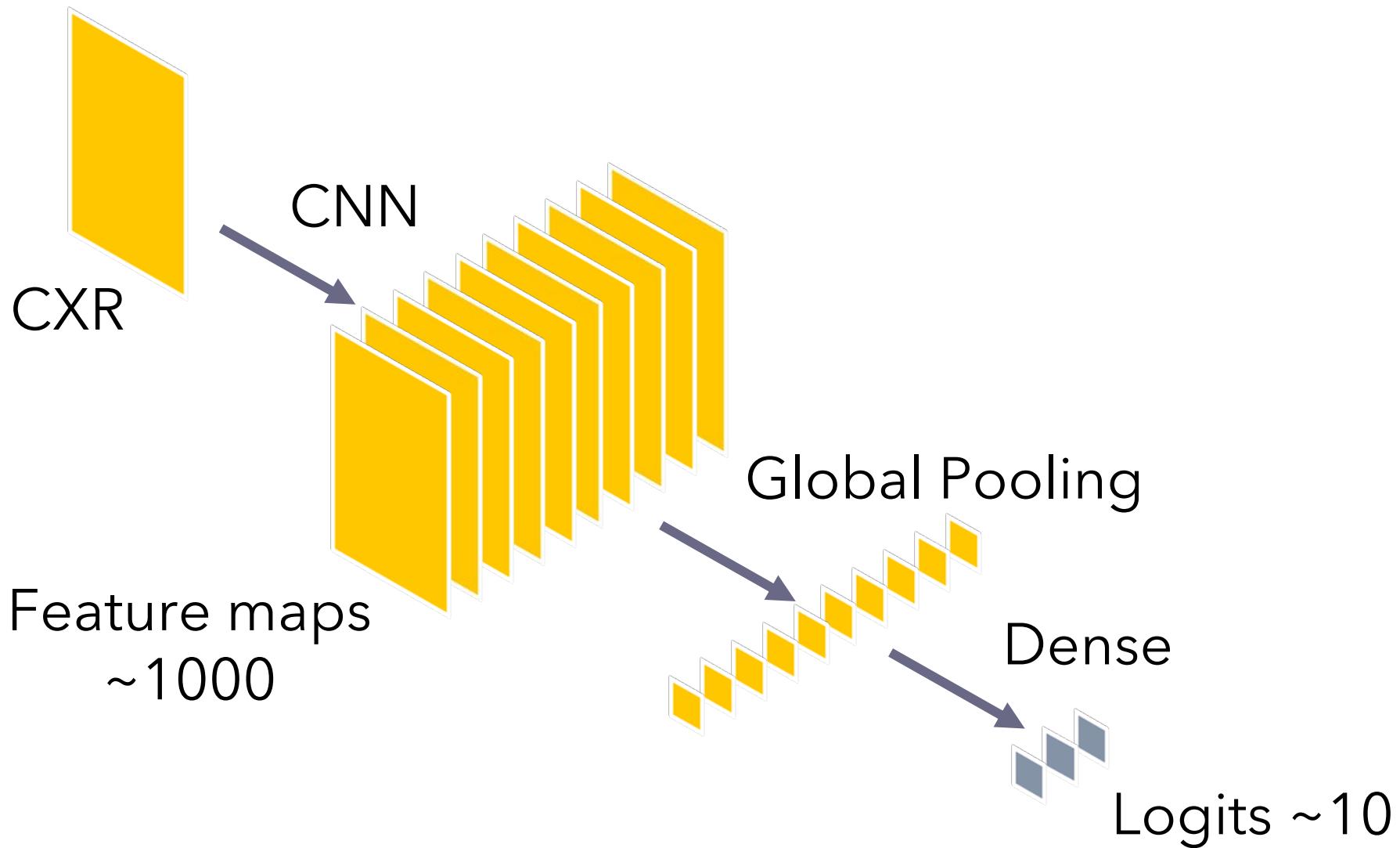


Predicted: syringe
(c)

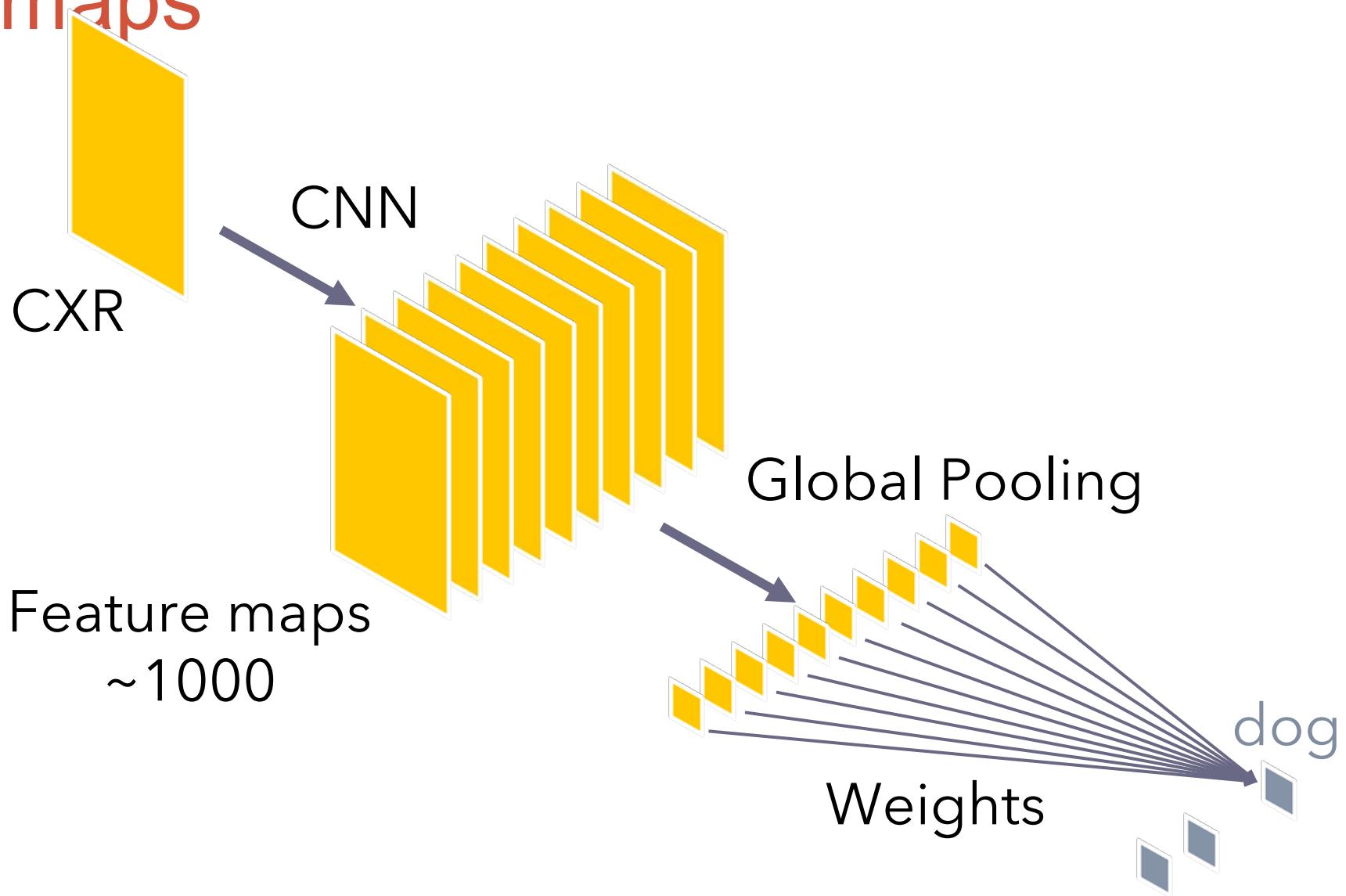


Predicted: vine snake
(d)

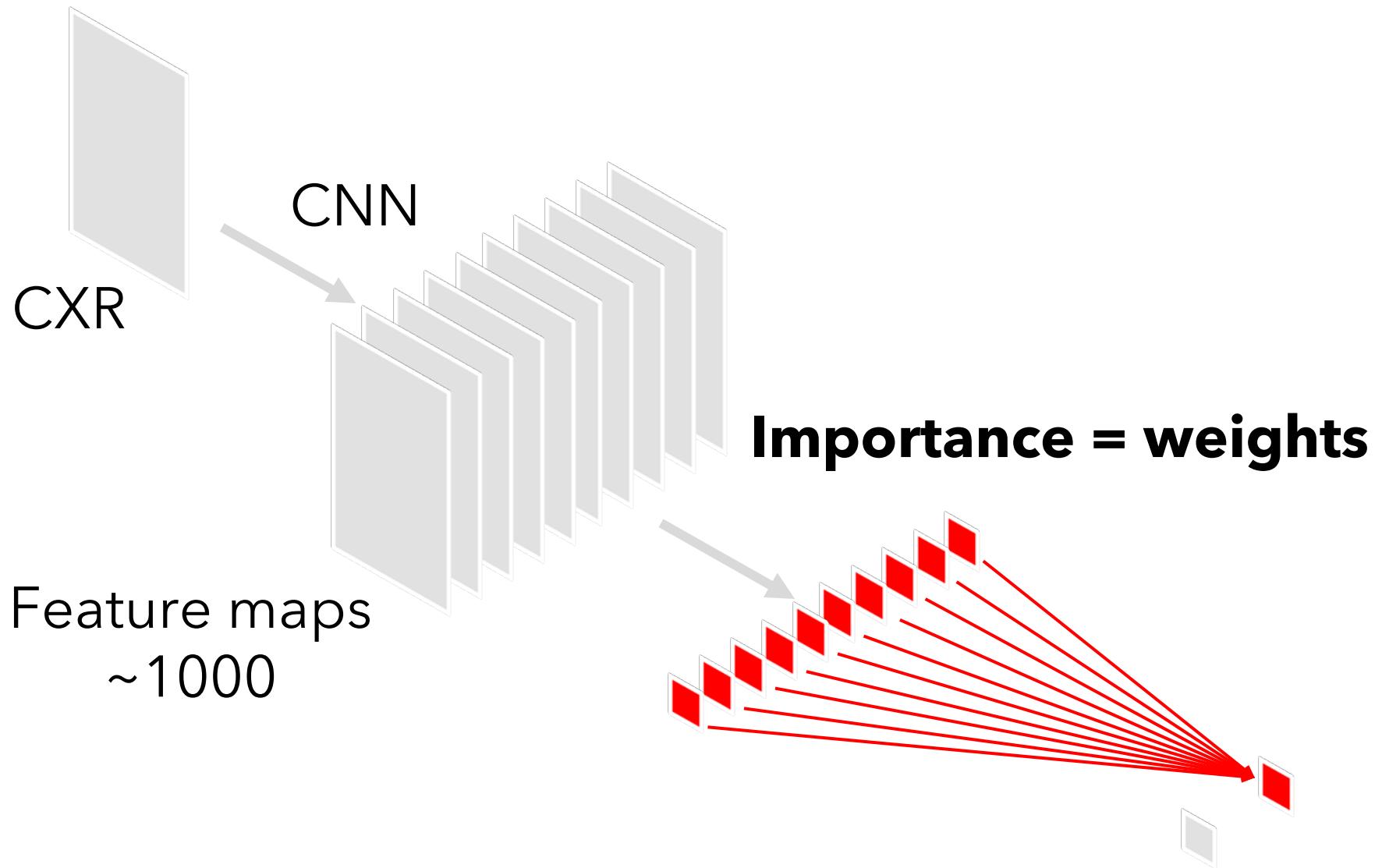
Typical CNN



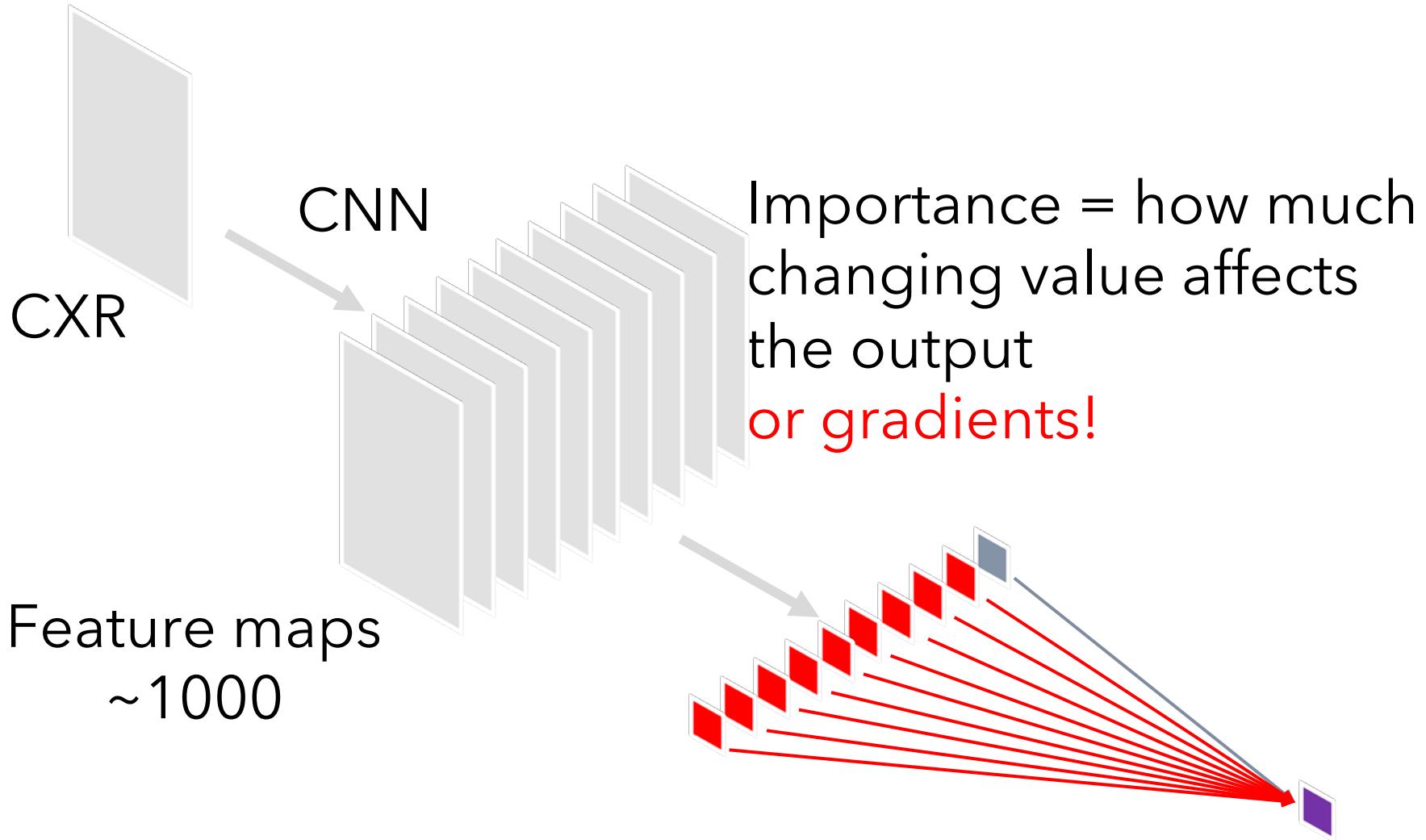
Prediction = weight sum of feat.
maps



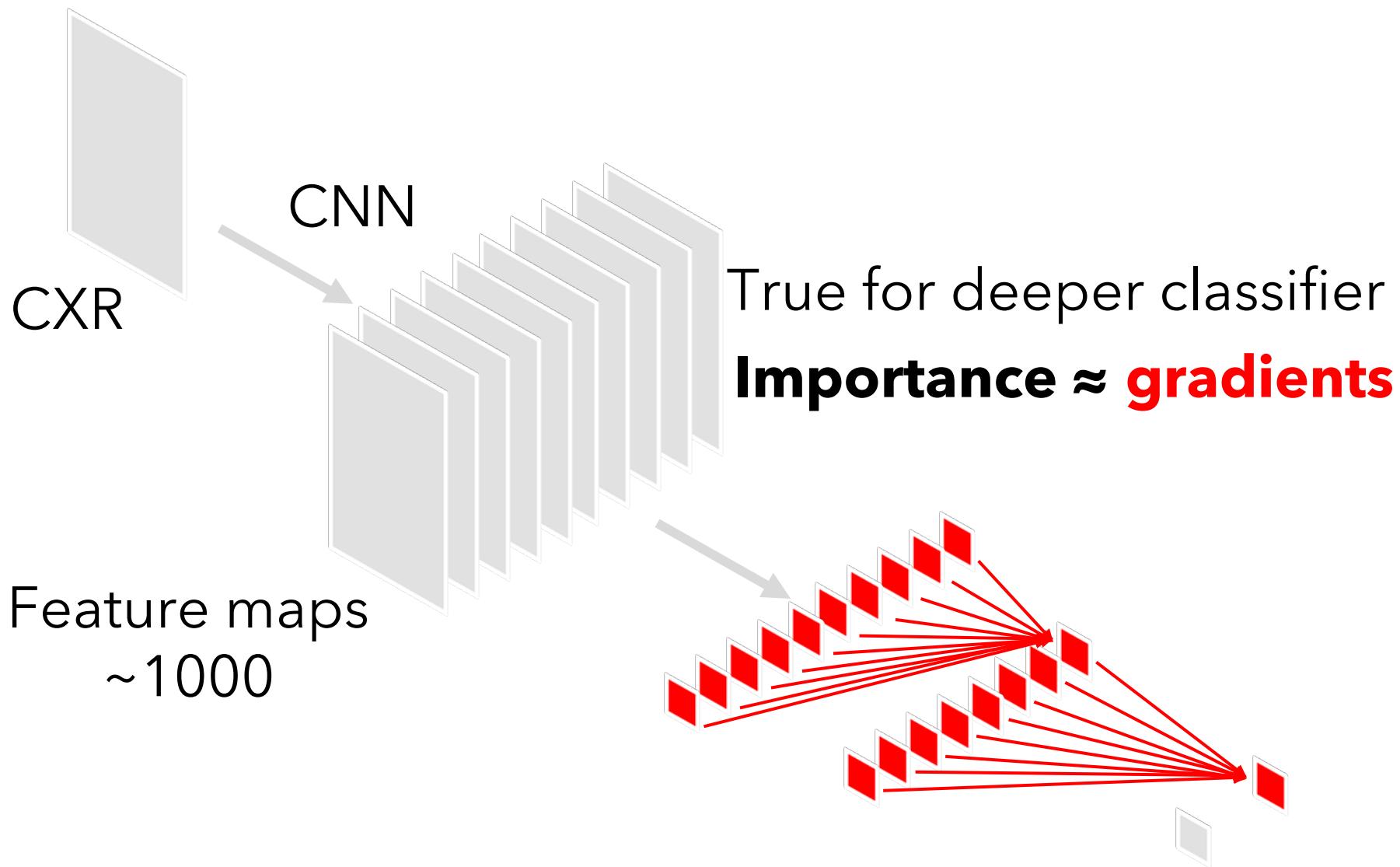
Feature importance = weights



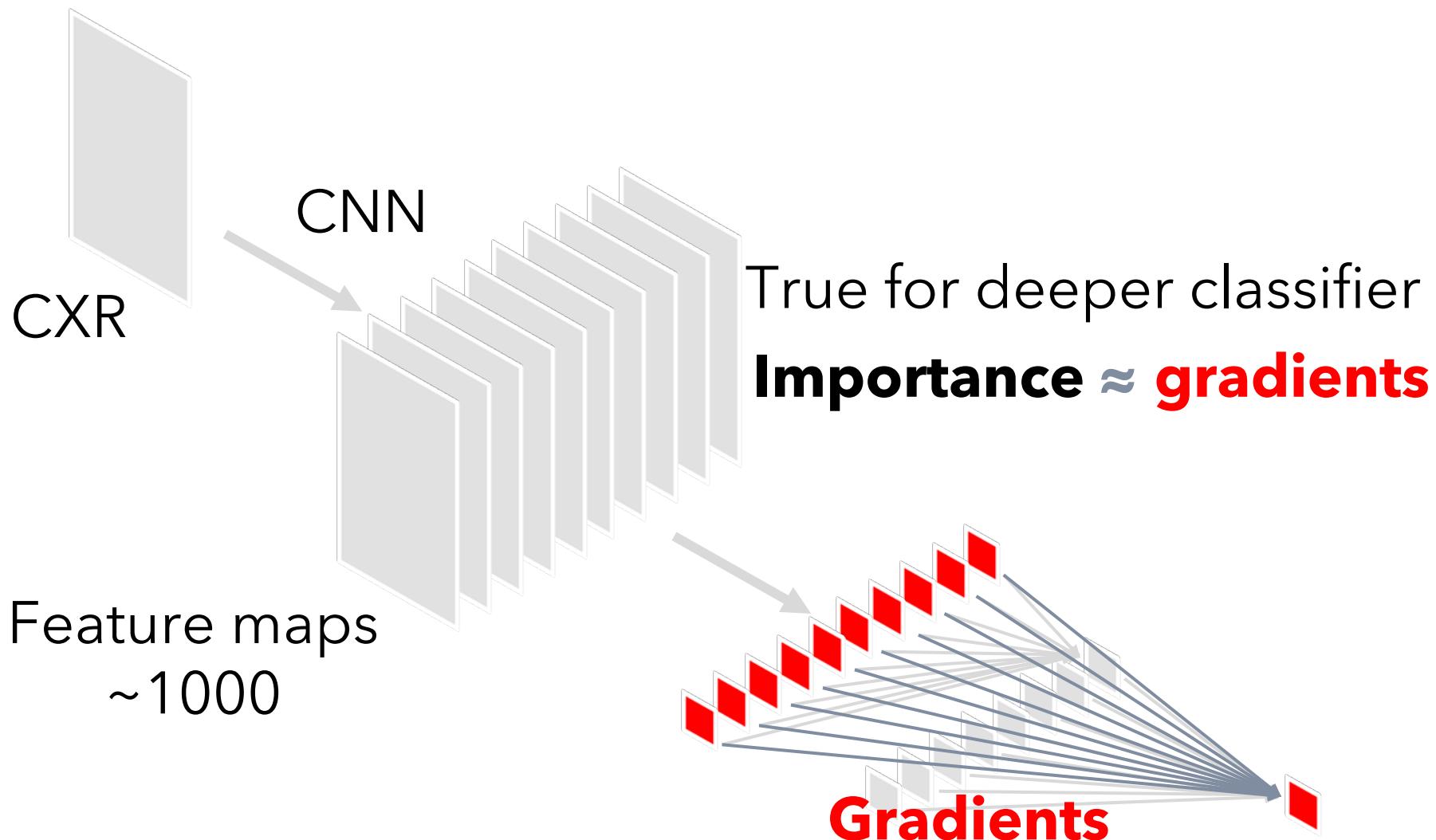
Feature importance ~ gradients



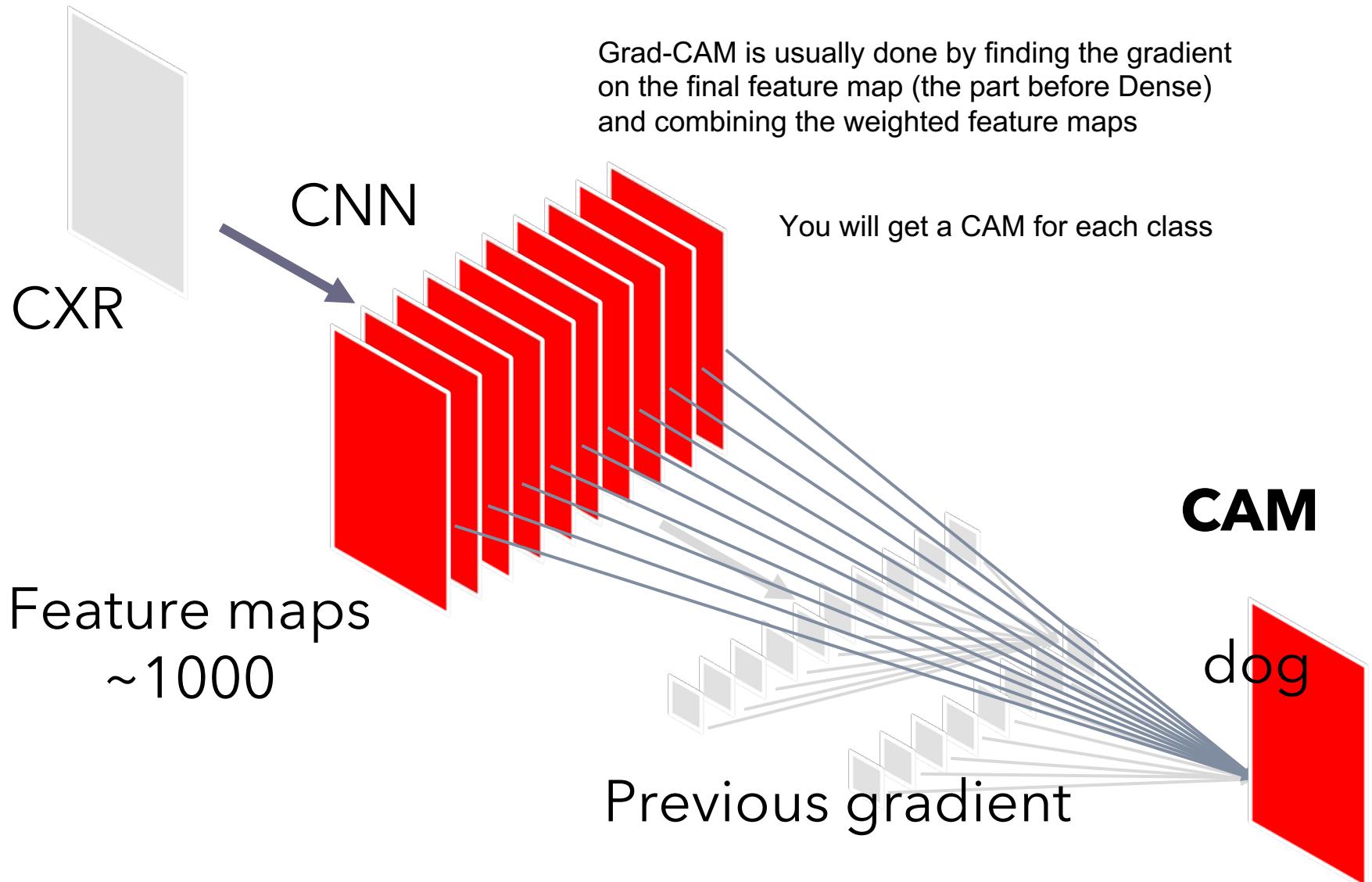
Feature importance \sim gradients



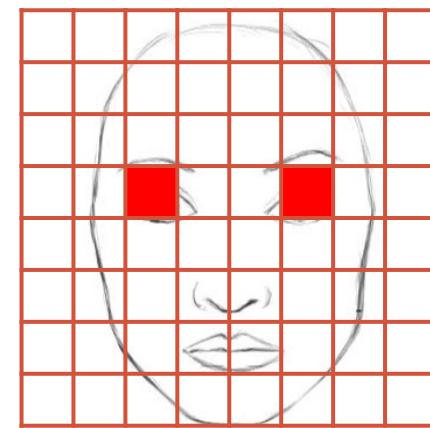
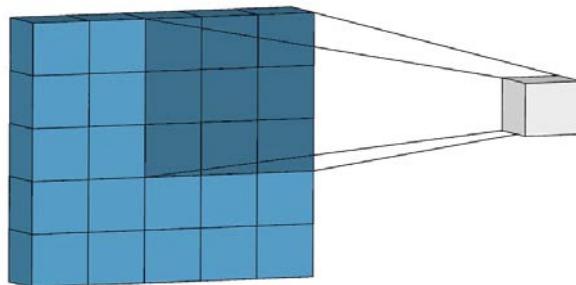
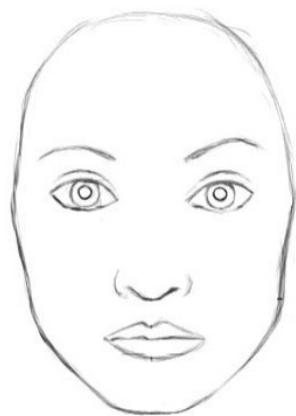
Feature importance ~ gradients



Class activation map (CAM)



Feature map



Eye feature map

<https://towardsdatascience.com/intuitively-understanding-convolutions-for-deep-learning-1f6f42faee1>

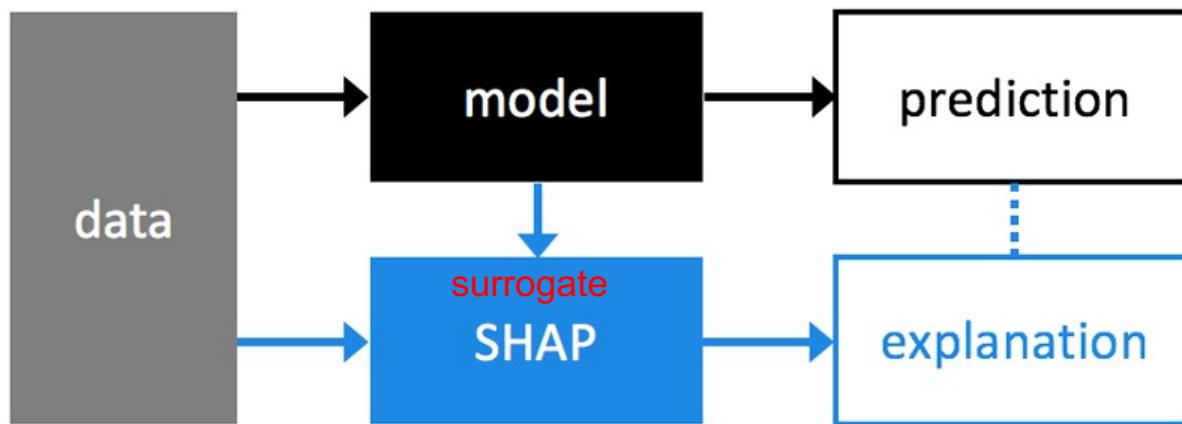
Output level: key ideas

A feature is important if I tweak the feature and the output change a lot. (sensitivity analysis)

Mostly useful for images types

Have a simpler model (**surrogate model**) explains the complicated model.

Converting things to logistic regression



Additive feature attribution

$$g(z') = \phi_0 + \sum_{i=1}^M \phi_i z'_i,$$

Simplified input features z' have binary values

z' can recover original features x , $x = h_x(z')$ (This mapping depends on x)

Goal: make $g(z') = f(h_x(z'))$. Then we can explain f in terms of simplified features. (**Solved by optimization**)

Example of simplified inputs

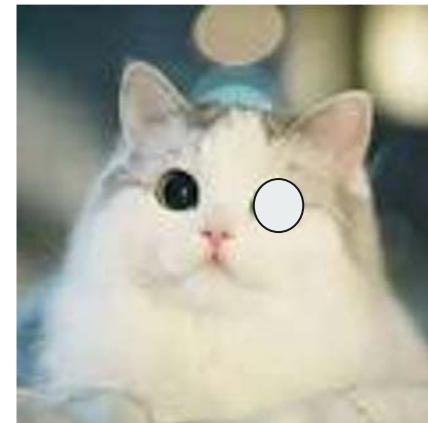
Original x = image

$z = 0$ if patch is not present

$z = 1$ if patch is present

$$h_x(z) = x \text{ if, } z=1$$

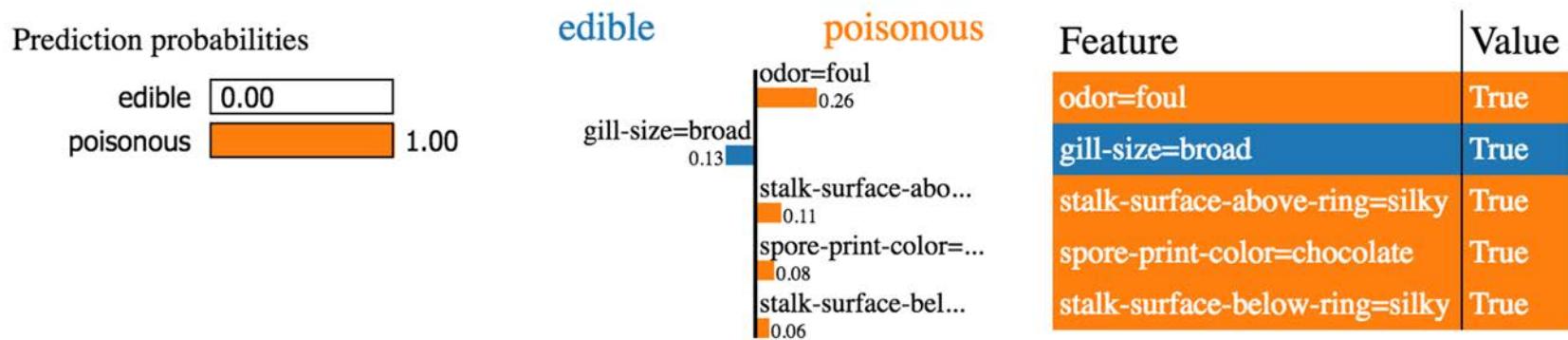
$$h_x(z) = x \text{ with missing patch, if } z=0$$



Additive feature attribution

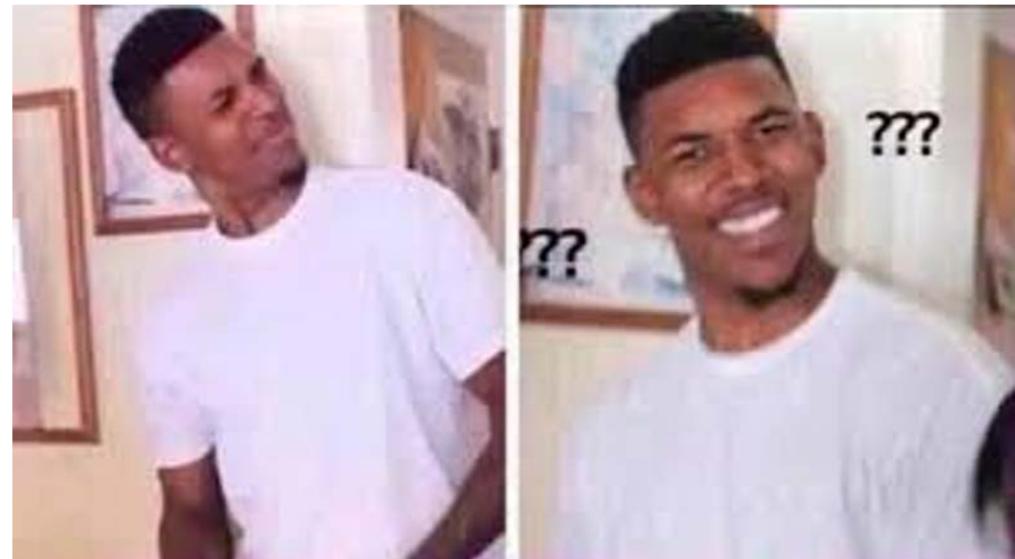
$$g(z') = \phi_0 + \sum_{i=1}^M \phi_i z'_i,$$

Many methods fall in additive feature attribution. For example, LIME



Introducing SHAP

SHAP is also an additive feature attribution, but gives credit to expected attribution



Example of simplified inputs

Original $x = \text{image}$

$z = 0$ if patch is not present

$z = 1$ if patch is present

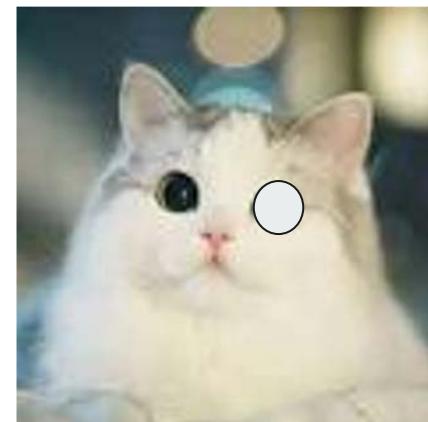
$$h_x(z) = x \text{ if, } z=1$$

$$h_x(z) = x \text{ with missing patch, if } z=0$$

This simplified input talks about this patch.

What happens if we change the other inputs?

Expected contribution talks about the contribution of the feature regardless of the other contributions

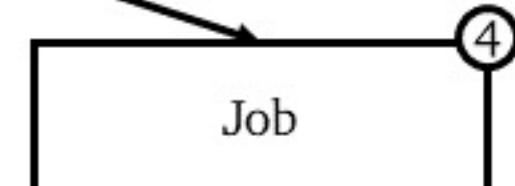
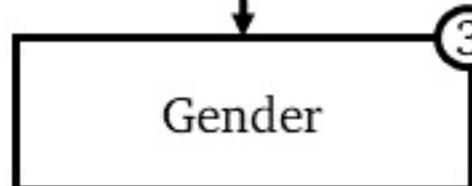
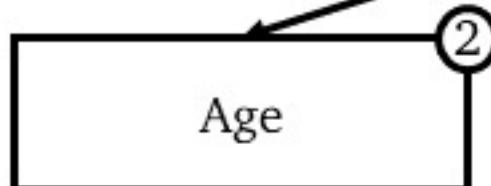


Let's say we have 3 features, and we want to create a model
We can create a power set of features, where we create a model for each set

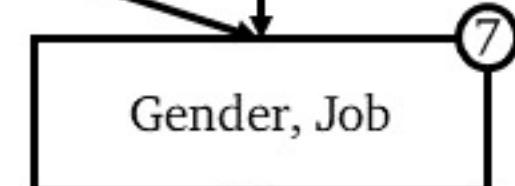
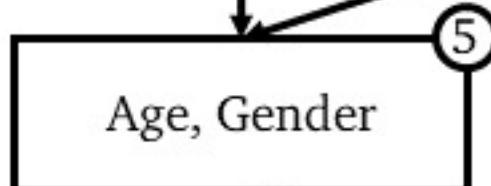
$f = 0$



$f = 1$



$f = 2$



$f = 3$

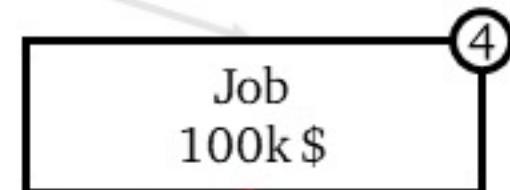


Expected contribution of Age feature is the average of adding age feature

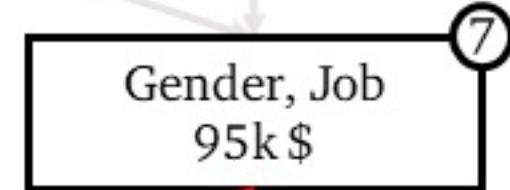
$f = 0$



$f = 1$



$f = 2$



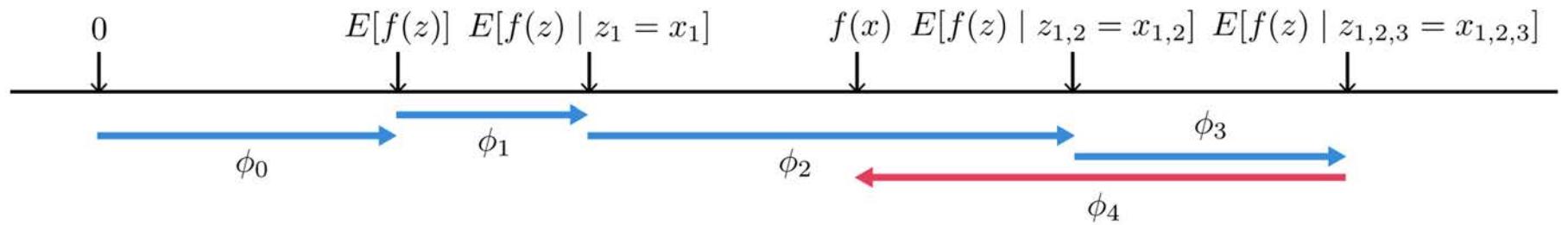
$f = 3$



Doing this exactly as shown is intractable in real tasks, SHAP employ tricks and approximations to do this.

<https://towardsdatascience.com/shap-explained-the-way-i-wish-someone-explained-it-to-me-ab81cc69ef30>

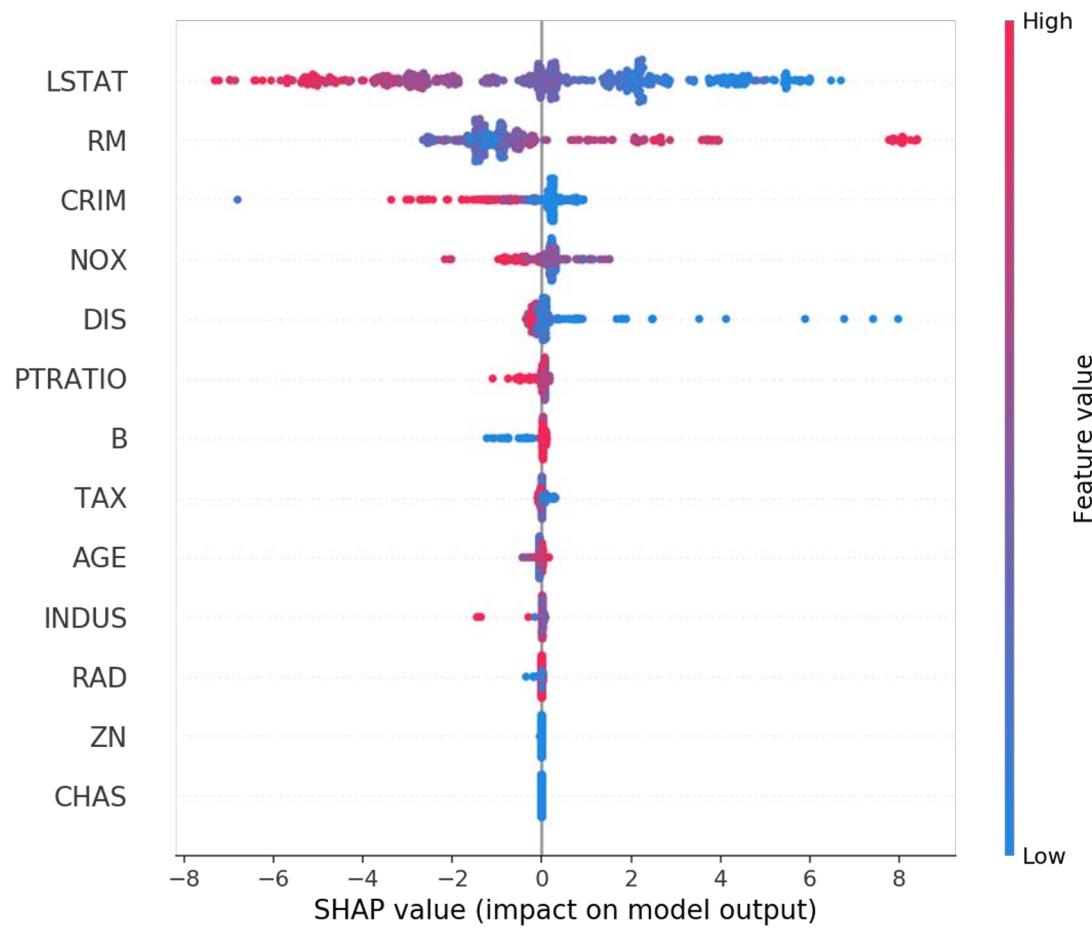
SHAP



SHAP example



SHAP example



Notes on additive attribution

If the features are correlated, it's hard to divide the attribution

Feature 1 = height

Feature 2 = height \ast 2

Notes on attributions

It does not necessarily tell how to improve.

If the explainer says the sales is low because of weak marketing

Does not mean increasing marketing will improve sales.

See “Causal Inference”

SHAP materials

- <https://github.com/slundberg/shap>
 - main repo
- <https://github.com/linkedin/fasttreeshap>
 - Fast SHAP for tree models
- A nice blog that explains SHAP computation
 - <https://towardsdatascience.com/shap-explained-the-way-i-wish-someone-explained-it-to-me-ab81cc69ef30>

Further readings

Rules of Machine Learning: Best Practices for ML Engineering

http://martin.zinkevich.org/rules_of_ml/rules_of_ml.pdf

Rules of Machine Learning: Best Practices for ML Engineering

Martin Zinkevich

This document is intended to help those with a basic knowledge of machine learning get the benefit of best practices in machine learning from around Google. It presents a style for machine learning, similar to the Google C++ Style Guide and other popular guides to practical programming. If you have taken a class in machine learning, or built or worked on a machine-learned model, then you have the necessary background to read this document.

[Rule #1: Don't be afraid to launch a product without machine learning.](#)

Ethics issues

- Abuse
- Bias

Deep fakes

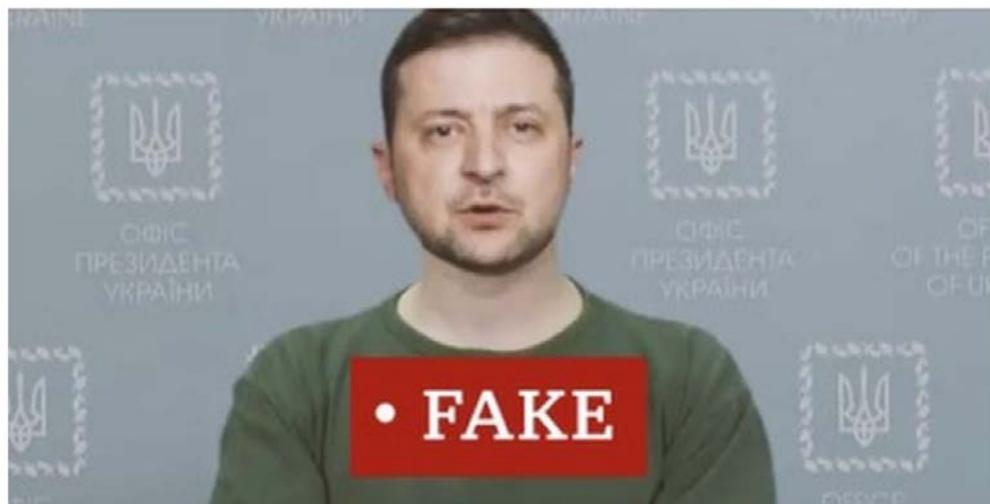
Deepfake presidents used in Russia-Ukraine war

By Jane Wakefield
BBC Technology

5 days ago



Russia-Ukraine war



<https://www.bbc.com/news/technology-60780142>

Fraudsters Used AI to Mimic CEO's Voice in Unusual Cybercrime Case

Scams using artificial intelligence are a new challenge for companies



PHOTO: SIMON DAWSON/BLOOMBERG NEWS

By Catherine Stupp

Updated Aug. 30, 2019 12:52 pm ET

<https://www.wsj.com/articles/fraudsters-use-ai-to-mimic-ceos-voice-in-unusual-cybercrime-case-11567157402>

MOST POPULAR NEWS

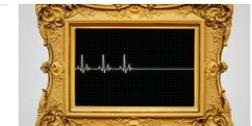
- Where State Abortion Laws Stand if Roe v. Wade Is Overturned



- They Kept Paying When Student Loan Debt Paused



- NFT Sales Are Flatlining



- Cerebral's Preferred Pharmacy Truepill Halts Adderall Prescriptions for All Customers



- Immigrants to Get Extension for Expiring or Expired U.S. Work



Tesla is updating Autopilot's 'Hold Steering Wheel' alert after complaints, says Elon Musk

Fred Lambert - Jun. 13th 2018 6:12 am ET  @FredericLambert



<https://electrek.co/2018/06/13/tesla-autopilot-hold-steering-wheel-alerts-complaints/>

'The Business of War': Google Employees Protest Work for the Pentagon



Thousands of Google employees have signed a letter to Sundar Pichai, the company's chief executive, protesting Google's role in a program that could be used to improve drone strike targeting.

Michael Short/Bloomberg

<https://www.nytimes.com/2018/04/04/technology/google-letter-ceo-pentagon-project.html>

Bias

- Our society is biased which leads to biased data and biased models
 - Understanding how models make prediction help us remove the bias
- Our morality is also biased
 - Understanding the cost-benefit tradeoff

Harms of Bias

Allocation

Models allocate resource to certain groups of people more - credit prediction

Representation

Models have more data of certain demographic and does better



<http://content.time.com/time/business/article/0,8599,1954643,00.html>

Study finds gender and skin-type bias in commercial artificial-intelligence systems

Examination of facial-analysis software shows error rate of 0.8 percent for light-skinned men, 34.7 percent for dark-skinned women.

Watch Video

Larry Hardesty | MIT News Office
February 11, 2018

▼ Press Inquiries

PRESS MENTIONS

<http://news.mit.edu/2018/study-finds-gender-skin-type-bias-artificial-intelligence-systems-0212>

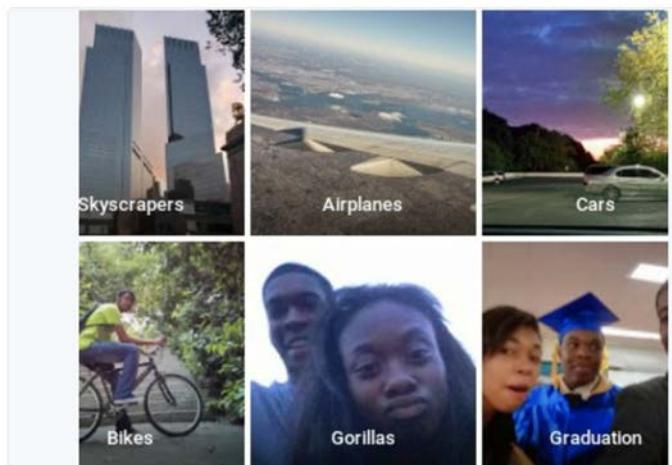
Ways to combat bias

- Improve accuracy <- hard
- Blacklist
- Equal representation
- Awareness
- Test the system on general users
- Try to de-bias <- if known bias, hard. If unknown bias, super hard

Google 'fixed' its racist algorithm by removing gorillas from its image-labeling tech

Nearly three years after the company was called out, it hasn't gone beyond a quick workaround

By James Vincent | @jjvincent | Jan 12, 2018, 10:35am EST



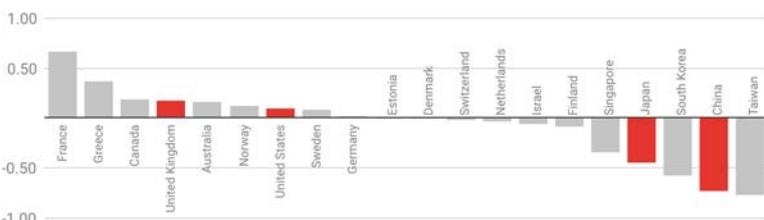
Jacky lives on @jalcine@playvicious.social now.
@jackyalcine

Google Photos, y'all fucked up. My friend's not a gorilla.
🕒 2,303 8:22 AM - Jun 29, 2015



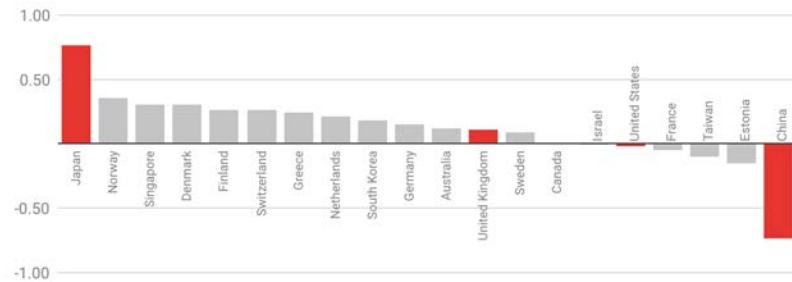
Moral machine

Countries with more individualistic cultures are more likely to spare the young



A comparison of countries piloting self-driving cars: If the bar is closer to 1, respondents placed a greater emphasis on sparing the young; if the bar is closer to -1, respondents placed a greater emphasis on sparing the old; 0 is the global average.

How countries compare in sparing pedestrians over passengers



If the bar is closer to 1, respondents placed a greater emphasis on sparing pedestrians; if the bar is closer to -1, respondents placed a greater emphasis on sparing passengers; 0 is the global average.

Problems with bias is both a social issue and a technological issue

<https://www.technologyreview.com/s/612341/a-global-ethics-study-aims-to-help-ai-solve-the-self-driving-trolley-problem/>

- “If I were to guess like what **our biggest existential threat** is, it’s probably that. So we need to be very careful with the artificial intelligence. There should be some regulatory oversight maybe at the national and international level, just to make sure that we don’t do something very foolish.”



- “I think people who are naysayers and try to drum up these doomsday scenarios — I just, I don’t understand it. It’s really negative and in some ways I actually think it is pretty irresponsible”

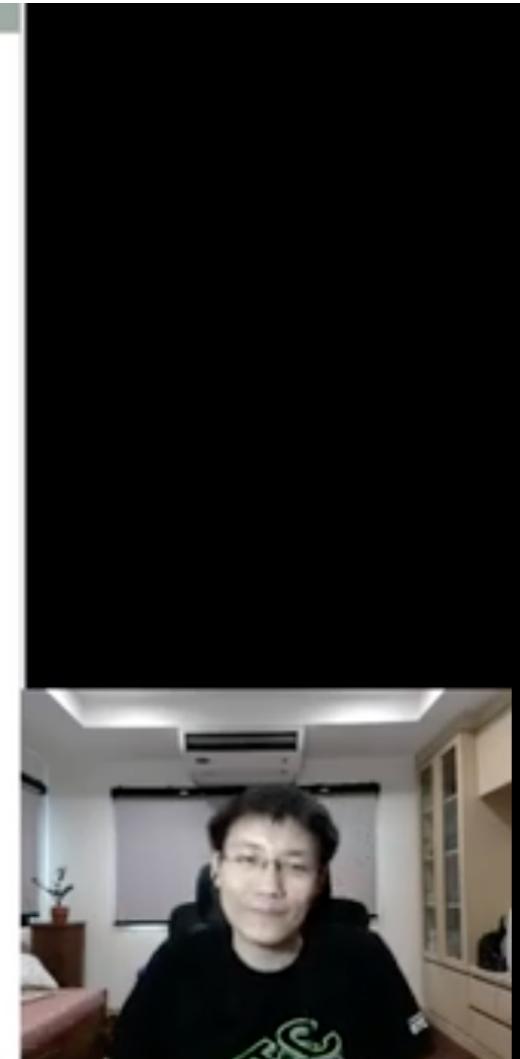


Poll



Poll

Elon 61 : Mark 39



Further Learning

- In Chula
 - NLP, ASR – more domain specific, less technical
- <https://www.youtube.com/c/YannicKilcher>
 - Funny and educational
- Follow your favorite researcher on Twitter
- <https://cds.nyu.edu/deep-learning/>
 - NYU has a nice advanced grad level deep learning course
 - Graph neural networks, energy-based models
- <https://www.bradyneal.com/causal-inference-course>
 - Causal inference
 - How to identify bias and de-bias?
- <https://stanford-cs329s.github.io/syllabus.html>
 - A more deployment-oriented course