Correlation, Model Combination

Jindřich Libovický (reusing materials by Milan Straka)

November 21, 2023

Charles University in Prague
Faculty of Mathematics and Physics
Institute of Formal and Applied Linguistics

Course Objectives: Where are we now?

After this course you should...

- Be able to reason about task/problems **suitable for ML**
  - Know when to use classification, regression and clustering
  - Be able to choose from this method Linear and Logistic Regression, Multilayer Perceptron, Nearest Neighbors, Naive Bayes, Gradient Boosted Decision Trees, k-means clustering
- Think about learning as (mostly probabilistic) **optimization on training data**
  - Know how the ML methods learn including theoretical explanation
- Know how to properly **evaluate** ML
  - Think about generalization (and avoiding overfitting)
  - Be able to choose a suitable evaluation metric
  - Responsibly decide what model is better
- Be able to **implement ML algorithms** on a conceptual level
- Be able to **use Scikit-learn** to solve ML problems in Python

Today's Lecture Objectives

After this lecture you should be able to

- Explain and implement different ways of measuring correlation: Pearson's correlation, Spearman's correlation, Kendall's τ.
- Decide if correlation is a good metric for your model.
- Measure inter-annotator agreement and draw conclusions for data cleaning and for limits of your models.
- Use correlation with human judgment to validate evaluation metrics.
- Ensemble models with uncorrelated predictions.
- Distill ensembles into smaller models.

Covariance

Given a collection of random variables $\mathrm{x}_1, \ldots, \mathrm{x}_N$, we know that

$$\mathbb{E}\left[\sum\nolimits_i \mathrm{x}_i\right] = \sum_i \mathbb{E}\big[\mathrm{x}_i\big].$$

But how about $\operatorname{Var}\left(\sum_i \mathrm{x}_i\right)$?

$$\begin{aligned}
\operatorname{Var}\left(\sum\nolimits_i \mathrm{x}_i\right) &= \mathbb{E}\left[\left(\sum\nolimits_i \mathrm{x}_i - \sum\nolimits_i \mathbb{E}[\mathrm{x}_i]\right)^2\right] \\
&= \mathbb{E}\left[\left(\sum\nolimits_i \big(\mathrm{x}_i - \mathbb{E}[\mathrm{x}_i]\big)\right)^2\right] \\
&= \mathbb{E}\left[\sum\nolimits_i \sum\nolimits_j \big(\mathrm{x}_i - \mathbb{E}[\mathrm{x}_i]\big)\big(\mathrm{x}_j - \mathbb{E}[\mathrm{x}_j]\big)\right] \\
&= \sum_i \sum_j \mathbb{E}\left[\big(\mathrm{x}_i - \mathbb{E}[\mathrm{x}_i]\big)\big(\mathrm{x}_j - \mathbb{E}[\mathrm{x}_j]\big)\right].
\end{aligned}$$

Covariance

We define **covariance** of two random variables x, y as

$$\operatorname{Cov}(\mathrm{x}, \mathrm{y}) = \mathbb{E}\Big[\big(\mathrm{x} - \mathbb{E}[\mathrm{x}]\big)\big(\mathrm{y} - \mathbb{E}[\mathrm{y}]\big)\Big].$$

Then,

$$\operatorname{Var}\left(\sum\nolimits_i \mathrm{x}_i\right) = \sum_i \sum_j \operatorname{Cov}(\mathrm{x}_i, \mathrm{x}_j).$$

Note that $\operatorname{Cov}(\mathrm{x}, \mathrm{x}) = \operatorname{Var}(\mathrm{x})$ and that we can write covariance as

$$\begin{aligned}\operatorname{Cov}(\mathrm{x}, \mathrm{y}) &= \mathbb{E}\Big[\big(\mathrm{x} - \mathbb{E}[\mathrm{x}]\big)\big(\mathrm{y} - \mathbb{E}[\mathrm{y}]\big)\Big] \\ &= \mathbb{E}\big[\mathrm{x}\mathrm{y} - \mathrm{x}\mathbb{E}[\mathrm{y}] - \mathbb{E}[\mathrm{x}]\mathrm{y} + \mathbb{E}[\mathrm{x}]\mathbb{E}[\mathrm{y}]\big] \\ &= \mathbb{E}\big[\mathrm{x}\mathrm{y}\big] - \mathbb{E}\big[\mathrm{x}\big]\mathbb{E}\big[\mathrm{y}\big].\end{aligned}$$

Correlation

Two random variables x, y are **uncorrelated** if $\operatorname{Cov}(\mathrm{x}, \mathrm{y}) = 0$; otherwise, they are **correlated**.

Note that two *independent* random variables are uncorrelated, because

$$\begin{aligned}
\operatorname{Cov}(\mathrm{x}, \mathrm{y}) &= \mathbb{E}\Big[\big(\mathrm{x} - \mathbb{E}[\mathrm{x}]\big)\big(\mathrm{y} - \mathbb{E}[\mathrm{y}]\big)\Big] \\
&= \sum_{x,y} P(x, y)\big(x - \mathbb{E}[x]\big)\big(y - \mathbb{E}[y]\big) \\
&= \sum_{x,y} P(x)\big(x - \mathbb{E}[x]\big) P(y)\big(y - \mathbb{E}[y]\big) \\
&= \sum_{x} P(x)\big(x - \mathbb{E}[x]\big) \sum_{y} P(y)\big(y - \mathbb{E}[y]\big) \\
&= \mathbb{E}_{\mathrm{x}}\big[\mathrm{x} - \mathbb{E}[\mathrm{x}]\big]\mathbb{E}_{\mathrm{y}}\big[\mathrm{y} - \mathbb{E}[\mathrm{y}]\big] = 0.
\end{aligned}$$

However, dependent random variables can be uncorrelated – random uniform x on $[-1, 1]$ and $\mathrm{y} = |\mathrm{x}|$ are not independent (y is completely determined by x), but they are uncorrelated.

Pearson correlation coefficient

There are several ways to measure correlation of random variables x, y.

Pearson correlation coefficient, denoted as ρ or r, is defined as

$$\begin{aligned} \rho &\stackrel{\text{def}}{=} \frac{\operatorname{Cov}(\mathrm{x}, \mathrm{y})}{\sqrt{\operatorname{Var}(\mathrm{x})}\sqrt{\operatorname{Var}(\mathrm{y})}} \\ r &\stackrel{\text{def}}{=} \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i (x_i - \bar{x})^2}\sqrt{\sum_i (y_i - \bar{y})^2}}, \end{aligned}$$

where:

- ρ is used when the full expectation is computed (population Pearson correlation coefficient);
- r is used when estimating the coefficient from data (sample Pearson correlation coefficient);
  - $\bar{x}$ and $\bar{y}$ are sample estimates of the respective means.

Pearson correlation coefficient

The value of Pearson correlation coefficient is in fact normalized covariance, because its value is always bounded by $-1 \leq \rho \leq 1$ (and the same holds for r).

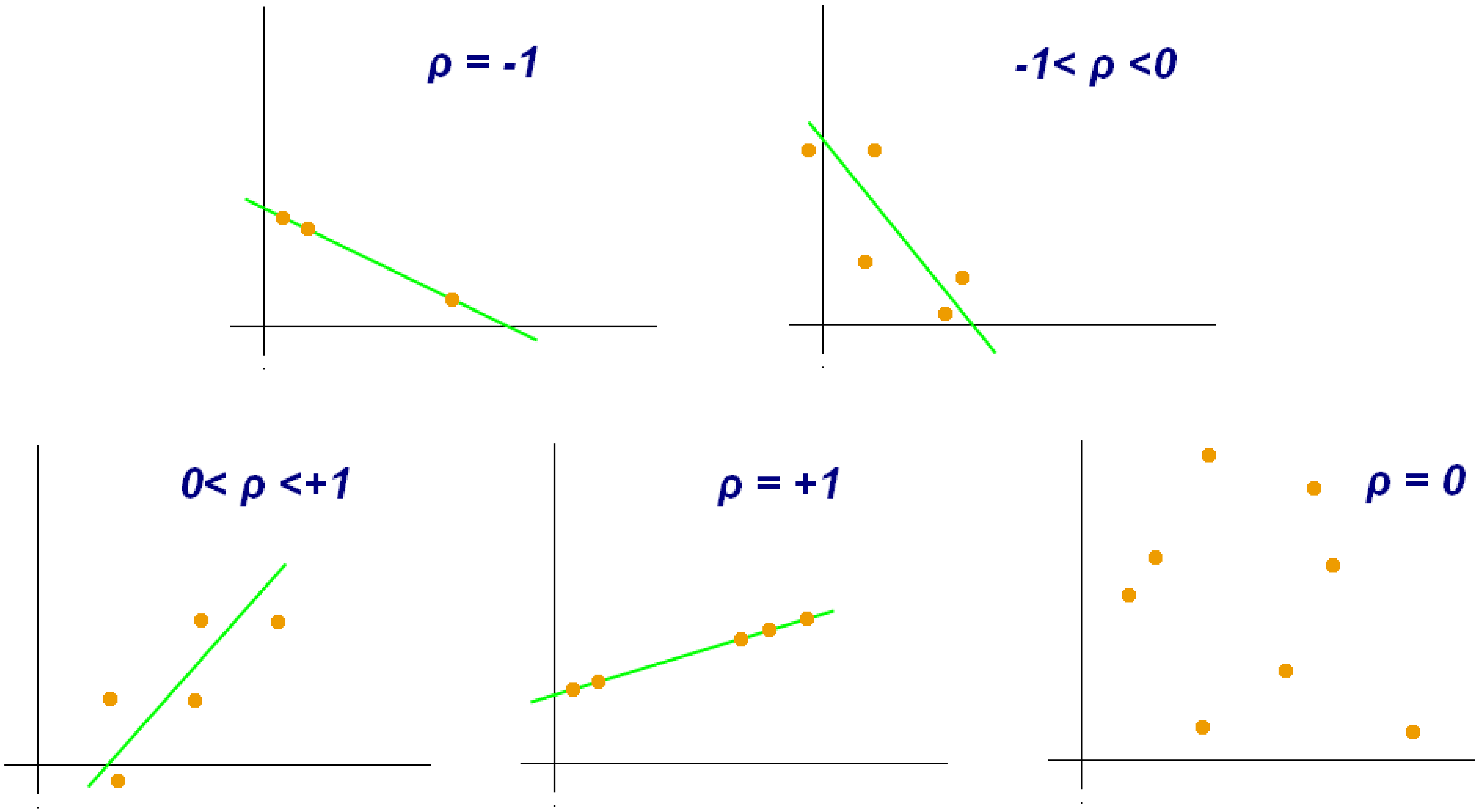
The bound can be derived from

$$\begin{aligned}
0 &\leq \mathbb{E}\left[\left(\frac{(\mathrm{x}-\mathbb{E}[\mathrm{x}])}{\sqrt{\mathrm{Var}(\mathrm{x})}} - \rho\frac{(\mathrm{y}-\mathbb{E}[\mathrm{y}])}{\sqrt{\mathrm{Var}(\mathrm{y})}}\right)^2\right] \\
&= \mathbb{E}\left[\frac{(\mathrm{x}-\mathbb{E}[\mathrm{x}])^2}{\mathrm{Var}(\mathrm{x})}\right] - 2\rho\mathbb{E}\left[\frac{(\mathrm{x}-\mathbb{E}[\mathrm{x}])}{\sqrt{\mathrm{Var}(\mathrm{x})}}\frac{(\mathrm{y}-\mathbb{E}[\mathrm{y}])}{\sqrt{\mathrm{Var}(\mathrm{y})}}\right] + \rho^2\mathbb{E}\left[\frac{(\mathrm{y}-\mathbb{E}[\mathrm{y}])^2}{\mathrm{Var}(\mathrm{y})}\right] \\
&= \frac{\mathrm{Var}(\mathrm{x})}{\mathrm{Var}(\mathrm{x})} - 2\rho\cdot\rho + \rho^2\frac{\mathrm{Var}(\mathrm{y})}{\mathrm{Var}(\mathrm{y})} = 1 - \rho^2,
\end{aligned}$$

which yields $\rho^2 \leq 1$.

Pearson correlation coefficient

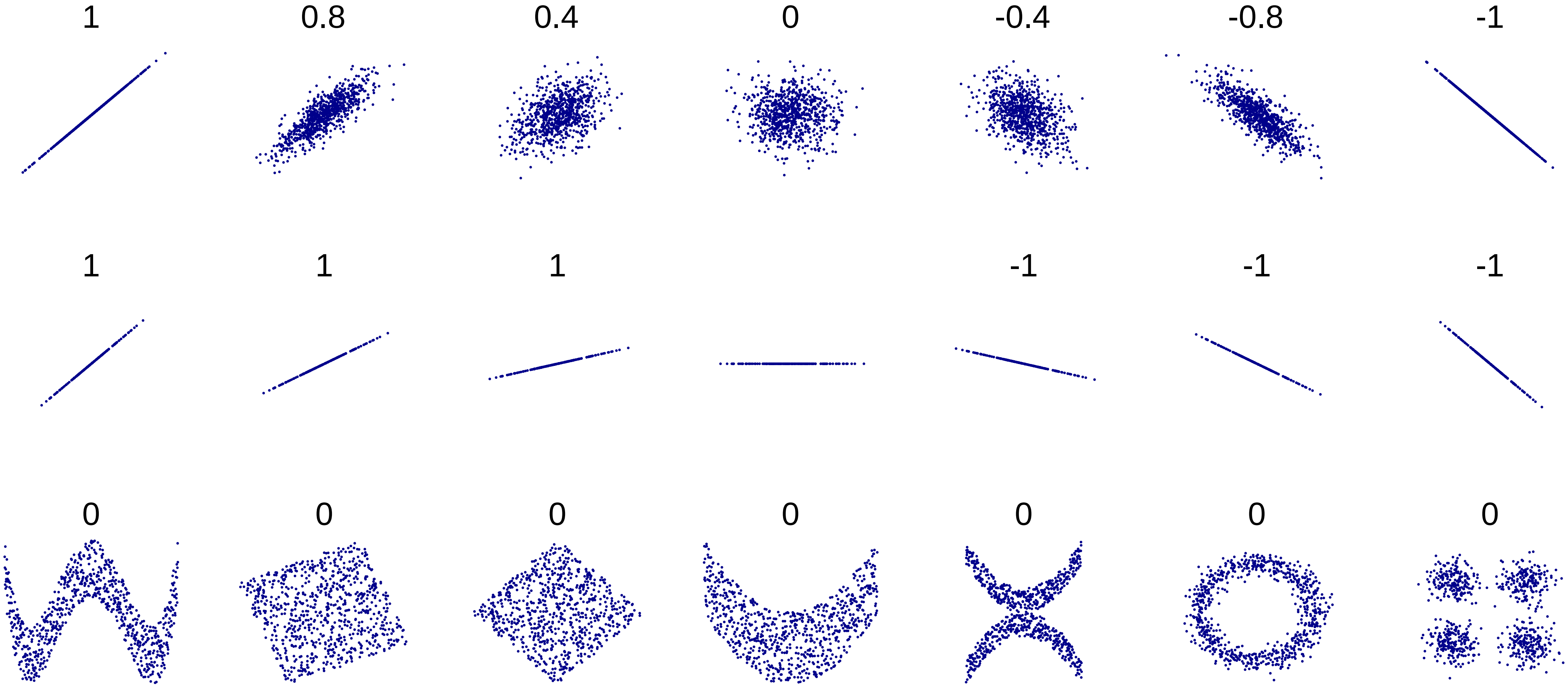
Pearson correlation coefficient quantifies **linear dependence** of the two random variables.



https://upload.wikimedia.org/wikipedia/commons/3/34/Correlation_coefficient.png

Pearson correlation coefficient

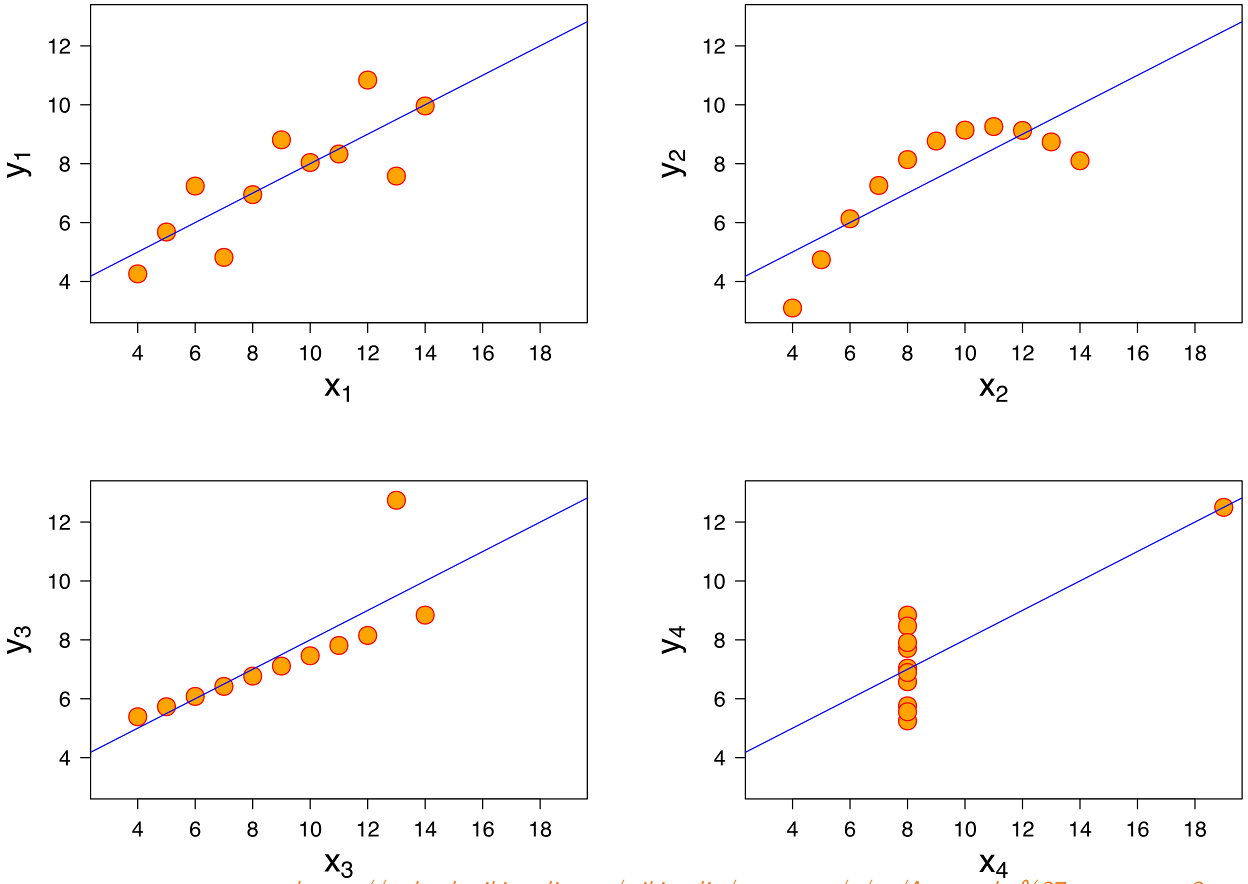
Pearson correlation coefficient quantifies **linear dependence** of the two random variables.



https://upload.wikimedia.org/wikipedia/commons/d/d4/Correlation_examples2.svg

Pearson correlation coefficient

The four displayed variables have the same mean 7.5, variance 4.12, Pearson correlation coefficient 0.816 and regression line $3 + \frac{1}{2}x$.



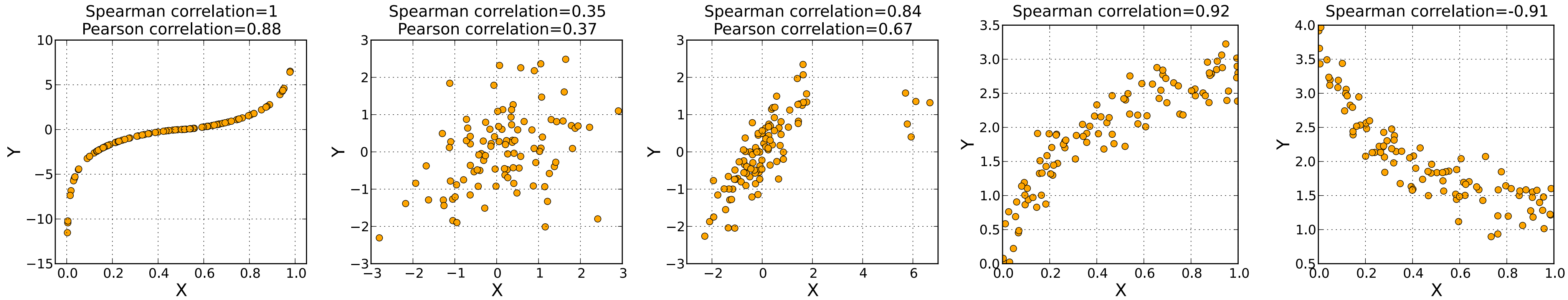
https://upload.wikimedia.org/wikipedia/commons/e/ec/Anscombe%27s_quartet_3.svg

Nonlinear Correlation – Spearman's ρ

To measure also nonlinear correlation, two coefficients are commonly used.

Spearman's rank correlation coefficient ρ

Spearman's ρ is Pearson correlation coefficient measured on **ranks** of the original data, where a rank of an element is its index in sorted ascending order.



https://upload.wikimedia.org/wikipedia/commons/4/4e/Spearman_fig{1,2,3,5,4}.svg

Kendall rank correlation coefficient τ

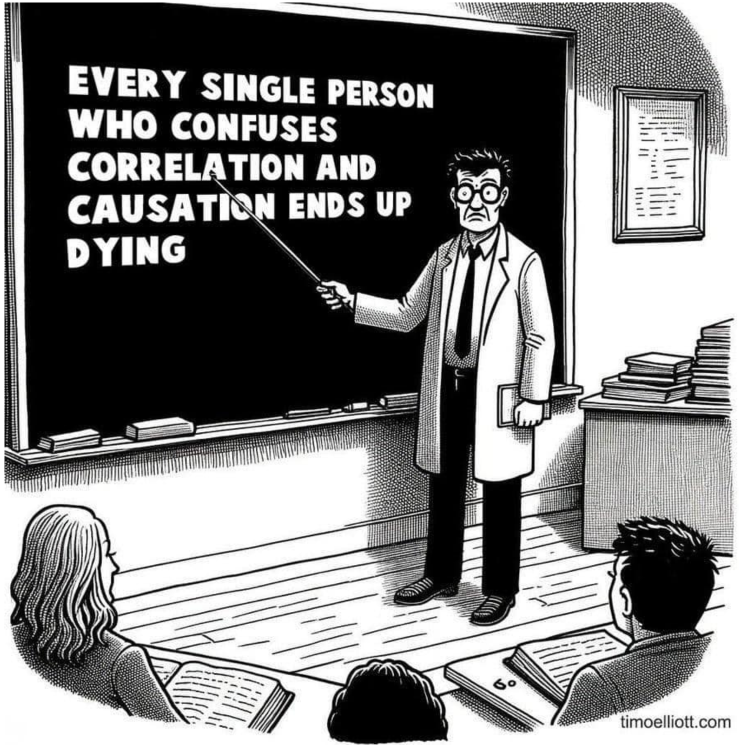
Kendall's τ measures the amount of *concordant pairs* (pairs where y increases/decreases when x does), minus the *discordant pairs* (where y increases/decreases when x does the opposite):

$$\begin{aligned}\tau &\overset{\text{def}}{=} \frac{\big|\{\text{pairs } i \neq j : x_j > x_i, y_j > y_i\}\big| - \big|\{\text{pairs } i \neq j : x_j > x_i, y_j < y_i\}\big|}{\binom{n}{2}} \\ &= \frac{\sum_{i<j} \operatorname{sign}(x_j - x_i) \operatorname{sign}(y_j - y_i)}{\binom{n}{2}}.\end{aligned}$$

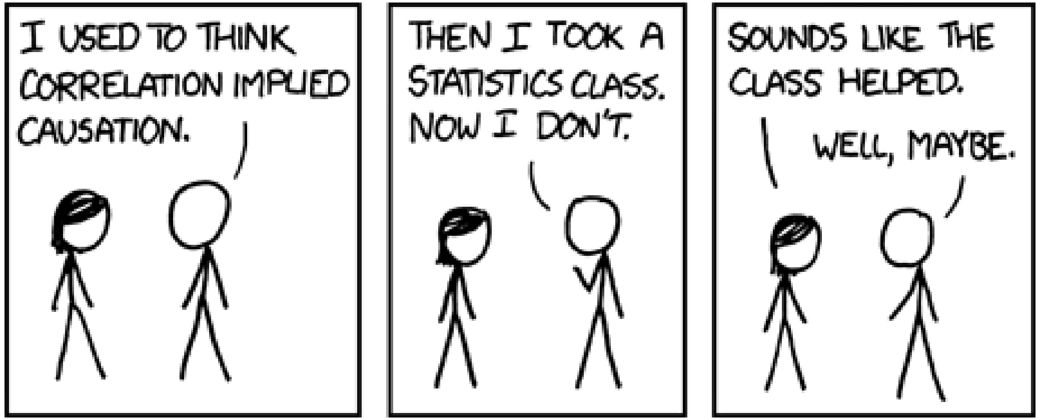
There is no clear consensus whether to use Spearman's ρ or Kendall's τ. When there are no/few ties in the data, Kendall's τ offers two minor advantages – $\frac{1+\tau}{2}$ can be interpreted as a probability of a concordant pair, and Kendall's τ converges to a normal distribution faster.

As defined, the range of Kendall's $\tau \in [-1, 1]$. However, if there are ties, its range is smaller – therefore, several corrections (not discussed here) exist to adjust its value in case of ties.

Correlation is not causation


EVERY SINGLE PERSON WHO CONFUSES CORRELATION AND CAUSATION ENDS UP DYING
timoelliott.com

https://timoelliott.com/blog/cartoons/yet-more-analytics-cartoons


I USED TO THINK CORRELATION IMPLIED CAUSATION.
THEN I TOOK A STATISTICS CLASS. NOW I DON'T.
SOUNDS LIKE THE CLASS HELPED.
WELL, MAYBE.

https://xkcd.com/552/

Use of Correlation in Machine Learning

In ML, correlation is commonly used as

- Evaluation metric for some tasks;
- Measuring data annotation quality;
- Assessing the quality of automatic metrics by comparing it to human judgment.

Correlation as evaluation metric

- Learning to rank (e.g., document retrieval): we do not care about the actual values
  - Kendall's τ, Spearman's correlation
  - When we want the correct items to rank before incorrect ones: precision (assuming fixed top-k, typically at 5, 10), recall (often ill-defined), mean reciprocal rank

$$\mathrm{MRR} = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{\text{rank of the first relevant item}}$$

- Evaluating pair similarity: word embeddings, sentence embeddings
  - Similarity estimates from psycholinguistic experiments: scores for word/sentence pairs
  - Measure Pearson/Spearman correlation between embedding distances and similarity scores

Inter-annotator agreement (1)

- Inter-annotator agreement can tell us
  - How well defined the task is
  - How reliable annotators/user ratings are
  - What data items are suspicious / difficult

- For continuous target values: Pearson's/Spearman's correlation
- For classification tasks: Cohen's κ

  p_O is observed agreement, p_E expected agreement by chance

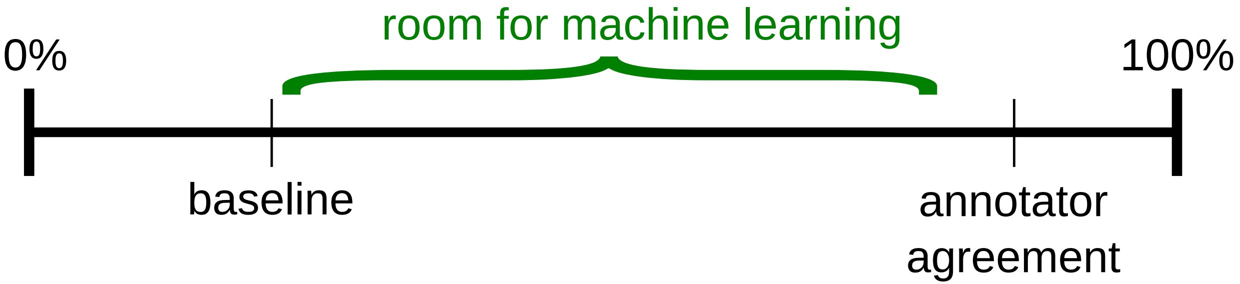
$$\kappa = \frac{p_O - p_E}{1 - p_E}$$

https://www.surgehq.ai/blog/the-pitfalls-of-inter-rater-reliability-in-data-labeling-and-machine-learning

Inter-annotator agreement (2)

- Can be used to filter out confusing data points and unreliable annotators
- Not all outliers are noise! Low IAA can reveal cultural differences.

IAA sets natural upper boundary for ML performance. Performance over IAA is suspicious!

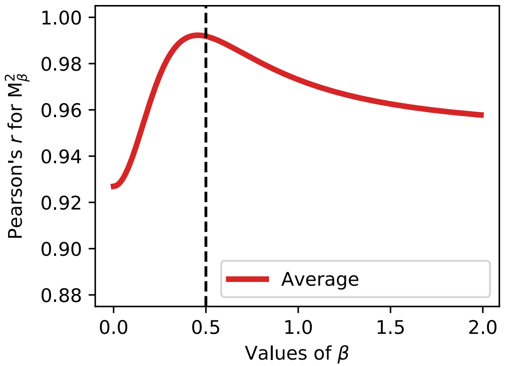


- Trivial baseline for classification: majority class, for regression average, or something based on simple rules
- Performance over IAA is more likely overfitting for the way the data is curated than super-human performance.

Correlation with human judgment

For some tasks, it might not be clear how to measure the model performance:

Grammar checking: the β parameter



J. Náplava, M. Straka, J. Straková, and A. Rosen. 2022. Czech Grammar Error Correction with a Large and Diverse Corpus. In TAACL, 10:452–467.

Machine translation: evaluation is subjective by definition, we design metrics to correlate with human judgment.

- SoTA machine translation metrics are typically machine-learned.
- Different metrics might be suitable different tiers of translation quality.
- There is an annual competition in MT quality and MT metric quality.

Model Combination aka Ensembling

The goal of **ensembling** is to combine several models in order to reach higher performance.

The simplest approach is to train several independent models and then to combine their predictions by averaging or voting.

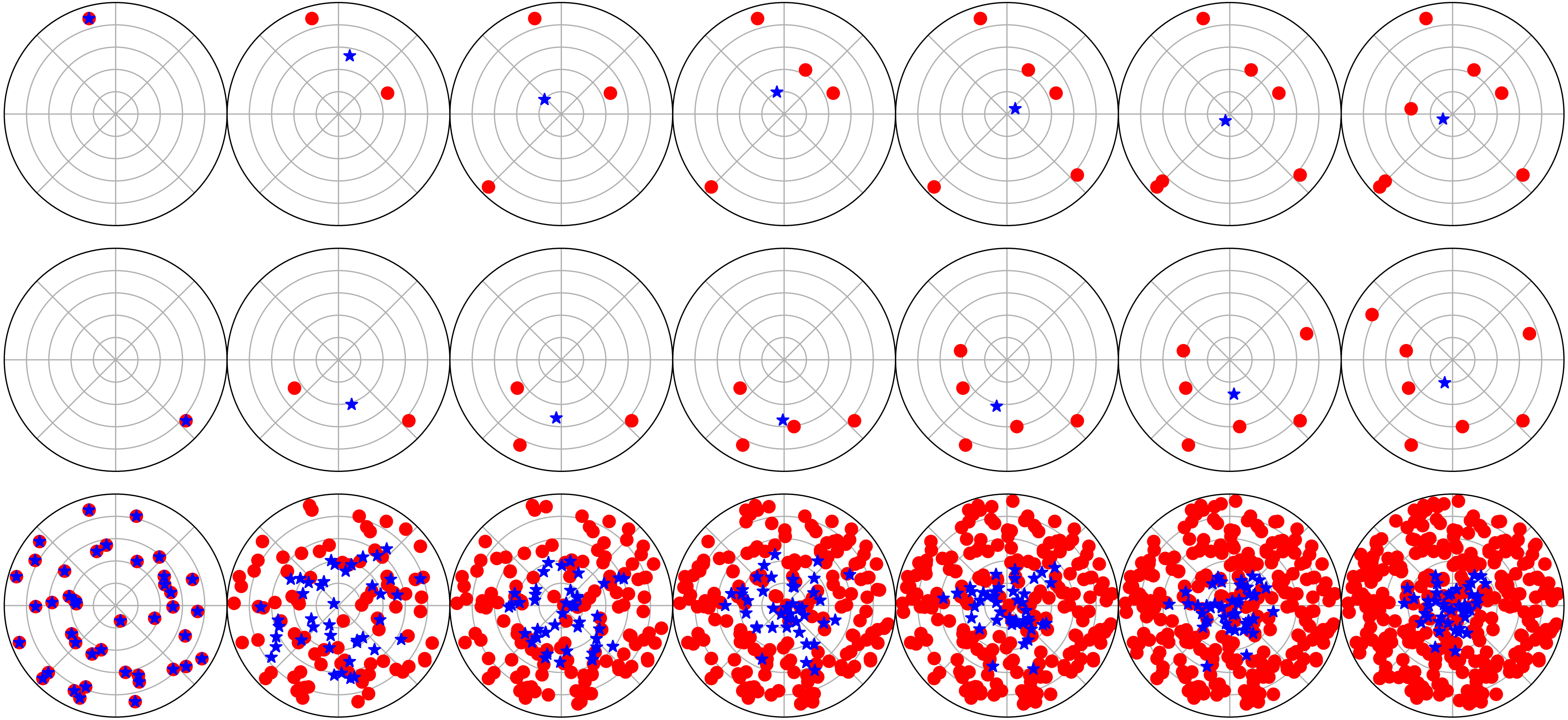
The terminology varies, but for classification:

- voting (or hard voting) usually means predicting the class predicted most often by the individual models,
- averaging (or soft voting) denotes averaging the returned model distributions and predicting the class with the highest probability.

The main idea behind ensembling is that if models have uncorrelated errors, then by averaging model predictions the errors will cancel out.

Visualization of Ensembling Performance

Consider ensembling predictions generated uniformly on a planar disc:



Model Combination aka Ensembling

If we denote the prediction of a model y_i on a training example $(\boldsymbol{x}, t)$ as $y_i(\boldsymbol{x}) = t + \varepsilon_i(\boldsymbol{x})$, so that $\varepsilon_i(\boldsymbol{x})$ is the model error on example $\boldsymbol{x}$, the mean square error of the model is

$$\mathbb{E}\big[(y_i(\boldsymbol{x}) - t)^2\big] = \mathbb{E}\big[\varepsilon_i^2(\boldsymbol{x})\big].$$

Considering M models, we analogously get that the mean square error of the ensemble is

$$\mathbb{E}\Bigg[\bigg(\frac{1}{M}\sum\nolimits_i \varepsilon_i(\boldsymbol{x})\bigg)^2\Bigg].$$

Finally, assuming that the individual errors ε_i have zero mean and are *uncorrelated*, we get that $\mathbb{E}\big[\varepsilon_i(\boldsymbol{x})\varepsilon_j(\boldsymbol{x})\big] = 0$ for $i \neq j$, and therefore,

$$\mathbb{E}\Bigg[\bigg(\frac{1}{M}\sum\nolimits_i \varepsilon_i(\boldsymbol{x})\bigg)^2\Bigg] = \mathbb{E}\Bigg[\frac{1}{M^2}\sum\nolimits_{i,j} \varepsilon_i(\boldsymbol{x})\varepsilon_j(\boldsymbol{x})\Bigg] = \frac{1}{M}\mathbb{E}\Bigg[\frac{1}{M}\sum\nolimits_i \varepsilon_i^2(\boldsymbol{x})\Bigg],$$

so the average error of the ensemble is $\frac{1}{M}$ times the average error of the individual models.

Bagging – Bootstrap Aggregation

For neural network models, training models with independent random initialization is usually enough, given that the loss has many local minima, so the models tend to be quite independent just when using different random initialization.

However, algorithms with convex loss functions usually converge to the same optimum independent of randomization.

In these cases, we can use **bagging**, which stands for **bootstrap aggregation**.

In bagging, we construct a different dataset for every model we train. We construct it using **bootstrapping** – we sample as many training instances as the original dataset has, but **with replacement**.

Such dataset is sampled using the same empirical data distribution and has the same size, but is not identical.

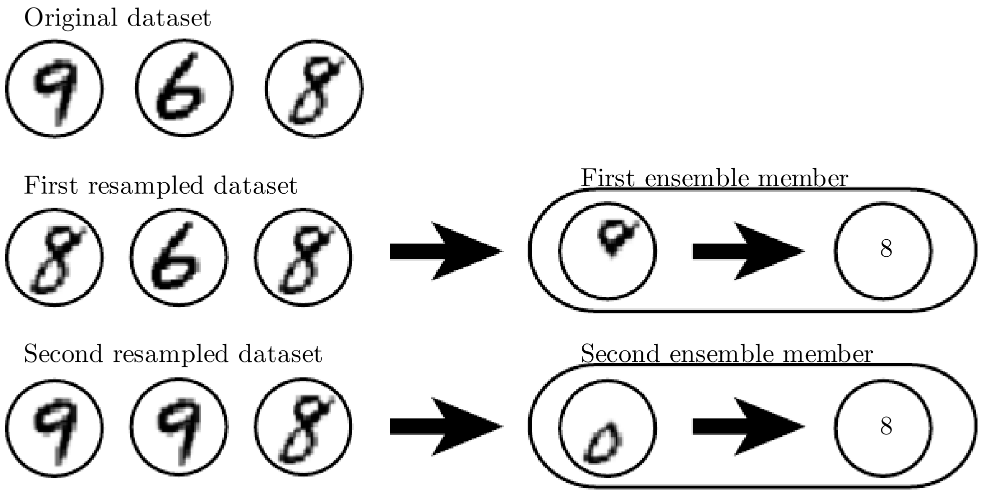


Figure 7.5 of "Deep Learning" book, https://www.deeplearningbook.org

Knowledge Distillation

- Model ensemble might be too slow or too big to use.
- Knowledge distillation $=$ training a **student** model that mimics behaviour of a **teacher model** (a bigger one or model ensemble).

Algorithm:

1. Process training data (or additional unlabelled data) with the best current model and get the output distribution $p_\textrm{teacher}(\boldsymbol y|\boldsymbol x; \boldsymbol w)$ (sometimes called *pseudolikelihood*)
2. Train a model with $H\left(p_\textrm{student}(\boldsymbol y|\boldsymbol x; \boldsymbol w), p_\textrm{teacher}(\boldsymbol y|\boldsymbol x; \boldsymbol w)\right)$ as a training objective.

Intuition: Complete distribution provides stronger supervision that just one-hot target, so it is easier for the smaller model to learn from such synthetic data.

Historical note: Term knowledge distillation come from a 2015 by Geoffrey Hinton et al., before a similar approach was called model compression.

Today's Lecture Objectives

After this lecture you should be able to

- Explain and implement different ways of measuring correlation: Pearson's correlation, Spearman's correlation, Kendall's τ
- Decide if correlation is a good metric for your model
- Measure inter-annotator agreement and draw conclusions for data cleaning and for limits of your models
- Use correlation with human judgment to validate evaluation metrics
- Ensemble models with uncorrelated predictions
- Distill ensembles into smaller models.