# Machine Learning for Health Care

Doruk Çetin

September 13, 2019

## Contents

# 1 Introduction

In biology, a **gene** is a sequence of nucleotides in DNA or RNA that codes for a molecule that has a function. During gene expression, the DNA is first copied into RNA. The RNA can be directly functional or be the intermediate template for a protein that performs a function.

The genotype is the part of the genetic makeup of a cell, and therefore of any individual, which determines one of its characteristics (phenotype). The **genotype–phenotype distinction** is drawn in genetics. "Genotype" is an organism's full hereditary information. "Phenotype" is an organism's actual observed properties, such as morphology, development, or behavior.

An **allele** is a variant form of a given gene. Sometimes, the presence of different alleles of the same gene can result in different observable phenotypic traits, such as different pigmentation. A **heterozygous** individual is someone who has two different alleles at a locus. For instance, using the sickle cell example, a heterozygous individual might have a genotype of AS. A **homozygous** individual has two identical alleles at a locus. The genotype for a homozygous individual might be AA or SS.

# 2 Support Vector Machines and Kernels for Computational Biology

## 2.1 Splicing

**RNA splicing**, in molecular biology, is a form of RNA processing in which a newly made precursor messenger RNA (pre-mRNA) transcript is transformed into a mature messenger RNA (mRNA). During splicing, **introns** are removed and **exons** are joined together.

Almost all donor splice sites exhibit GU and almost all acceptor sites exhibit AG, but not all GUs and AGs are used as splice site.

## 2.2 Intro. to ML, SVMs, Kernels

Hyperplane classifiers with large margins have small VC dimension. Maximum margin means minimum complexity.

**Kernel trick**: Scalar product in feature space can be computed in input space.

Common kernels: polynomial, sigmoid, RBF, normalization...
What can you do if kernel is not positive definite?

- Optimization problem is not convex
- Add constant to diagonal (cheap)
- Exponentiate kernel matrix (all eigvals become positive)
- SVM-pairwise use similarity as features

## 2.3 String Kernels

Kernels allow to encode application-specific knowledge. Many kernels for different applications available.

General ides: Count substrings shared by two strings. The greater the number of common substrings, the more two sequences are deemed similar.

True sites: fixed window around a true site.

String kernel SVMs capable of efficiently dealing with large k-mers k ¿ 10.

**Spectrum kernel**

- Makes use of **position-independent motifs**
- Like bag-of-words kernel for text classification
- Count k-mers in both sequences
- Spectrum Kernel is sum of product of counts (for same k-mer)
- **Spectrum Kernel with Mismatches**: Do not enforce strictly exact matches

**Weighted-degree kernel**

- Makes use of **position-dependent motifs**
- Mixture of spectrum kernels (up to degree d)
- Each position is considered independently
- As weighting use $\beta_k = 2\frac{d-k+1}{d(d+1)}$, where d is the maximal match length taken into account. This way the longer matches are weighted less, but they imply many shorter matches.
- Naive computation is $O(L \cdot d)$, block formulation reduces the cost to $O(L)$

**Weighted Degree Kernel with Shifts**

- Makes use of **partially position-dependent motifs**
- If sequence is slightly mutated (e.g. indels), WD kernel fails. (Indel is a molecular biology term for an insertion or deletion of bases in the genome of an organism.)
- Extension: Allow some positional variance (shifts S(l))

**Fisher & TOP Kernels**

- Combine probabilistic models and SVMs.
- Fisher Kernel on PSSM is identical to WD kernel with order 1.
- Marginalized-count kernels can be understood as a generalization of Fisher kernels.

## 2.4 Extracting Insights from SVMs

SVM decision function is $\alpha$-weighting of training points, but we are interested in weights of features. We can explicitly compute w and use it to rank importance.

SVM scoring function:

- Explicit representation of w allows (some) interpretation
- SVM-w does not reflect the score for a motif as substrings and overlapping strings contribute, too!

**SVM drawbacks**

- No confidence of predictions provided
- No generative model
- Prior knowledge may be difficult to encode as kernel

**Positional Oligomer Importance Matrices (POIMs)**

- Given k-mer z at position j in the sequence, compute expected score $\mathbb{E}[s(x) \mid x[j] = z]$ (for small k)
- Normalize with expected score over all sequences
- For large k use differential POIM

The lowest order POIM (k=1) essentially conveys the same information as is represented in a sequence logo. However, unlike sequence logos, POIMs naturally generalize to higher order nucleotide patterns.

## 2.5 Position weight matrix

A position weight matrix (PWM), also known as a position-specific weight matrix (PSWM) or **position-specific scoring matrix (PSSM)**, is a commonly used representation of motifs (patterns) in biological sequences. PWMs are often derived from a set of aligned sequences that are thought to be functionally related and have become an important part of many software tools for computational motif discovery.

A PWM has one row for each symbol of the alphabet: 4 rows for nucleotides in DNA sequences or 20 rows for amino acids in protein sequences. It also has one column for each position in the pattern. In the first step in constructing a PWM, a basic position frequency matrix (PFM) is created by counting the occurrences of each nucleotide at each position. From the PFM, a position probability matrix (PPM) can now be created by dividing that former nucleotide count at each position by the number of sequences, thereby normalising the values. Most often the elements in PWMs are calculated as log likelihoods.

# 3 Biomedical Natural Language Processing

**Also see related sections on both CIL and NLU notes. Notes on word embeddings are currently collected in a separate file.**

Problems with clinical texts:

- Ungrammatical, has misspellings and concatenations. Contains short telegraphic phrases, acronyms, abbreviations, which are often overloaded.
- It can contain many things that can be typed or pasted, such as long sets of lab values or vital signs.
- Idiosyncratic and institution-specific template-use is common.
- Pervasive fear, misunderstanding, and confusion around security, privacy, de-identification, and anonymization = ridiculous amount of agony in getting access

## 3.1 Ontologies

Definition: a set of concepts and categories in a subject area or domain that shows their properties and the relations between them.

Semantic way of representing knowledge of the domain. Intelligent system can provide reasoning systems to make inferences within the ontology.

Two main objectives: share the common structure of information, reuse the similar ontology in another domain

BioMedical ontologies: OpenCyc, WordNet, Galen, **UMLS (Unified Medical Language System)**, FMA, SNOMED-CT, Gene Ontology

Metathesaurus: synonymous terms clustered into a concept, preferred term is chosen, unique identifier (CUI) is assigned.

Semantic network: semantic types and semantic relationships form the structure of the network, broadly categorize the biomedical domain.

## 3.2 Text Processing

- Given a character sequence and a defined document unit, **tokenization** is the task of chopping it up into pieces, called tokens, perhaps at the same time throwing away certain characters, such as punctuation.
- **Token normalization** is the process of canonicalizing tokens so that matches occur despite superficial differences in the character sequences of the tokens. The most standard way to normalize is to implicitly create equivalence classes, which are normally named after one member of the set. (Asymmetric expansion illustrates a shortcoming of the equivalence class approach, which can be improved through expansion lists.)
- **Stemming** usually refers to a crude heuristic process that chops off the ends of words in the hope of achieving this goal correctly most of the time, and often includes the removal of derivational affixes.
- **Lemmatization** usually refers to doing things properly with the use of a vocabulary and morphological analysis of words, normally aiming to remove inflectional endings only and to return the base or dictionary form of a word, which is known as the **lemma**.

## 3.3 Bag-of-words model

**Term frequency (TF)**: the raw count of a term in a document: the number of times that term t occurs in document d. TF suffers from a critical problem: all terms are considered equally important. In fact certain terms have little or no discriminating power in determining relevance. Basic formulation is $f_{t,d} / \sum_{t'} f_{t',d}$

**Document Frequency (DF)**: the number of documents in the collection that contain a term t. Basic formulation of IDF is $\log N/n_t$

In information retrieval, tf–idf or TFIDF, short for **term frequency–inverse document frequency**, is a numerical statistic that is intended to reflect how important a word

is to a document in a collection or corpus. It is often used as a weighting factor in searches of information retrieval, text mining, and user modeling.

**n-grams**: Bag-of-word model is an order less document representation. One way to overcome this problem is to take n number of terms as a unit.

## 3.4 Part-of-speech tagging

**Brill tagger** (Transformation-Based Learning): a transformation-based process, in the sense that a tag is assigned to each word and changed using a set of predefined rules. In the transformation process, if the word is known, it first assigns the most frequent tag, or if the word is unknown, it naively assigns the tag "noun" to it. Applying over and over these rules, changing the incorrect tags, a quite high accuracy is achieved. This approach ensures that valuable information such as the morphosyntactic construction of words is employed in an automatic tagging process.

## 3.5 Word vectors

Discrete representations: use a taxonomy that has hypernyms (is-a) relationships and synonym sets (e.g. WordNet).
Problems with discrete representations:

- Missing new words - hard to keep up with the evaluation of the language.
- Requires human labor to create and adapt.
- Subjective.
- Hard to compute accurate word similarity.
- Dimensionality problems (one-hot encoding).

Problems with simple co-occurrence vectors:

- Increase in size with vocabulary. Very high dimensional: require a lot of storage.
- Subsequent classification models have sparsity issues. Less robust models as a result.
- Dimensionality reduction with SVD: Computational cost scales quadratically for $n * m$ matrix: $O(mn2)$, hard to incorporate new words or documents

**Why use embeddings?**

- Reduce dimensionality of representation.
- Encodes similarity information, useful for other tasks.
- Learn representations of entities (words) as well as relationships between them.

**Word2Vec**

- Train a classifier on a binary prediction task of words occurring in the neighbourhoods of other words, take the learned classifier weights as the word embeddings.
- Faster and can easily incorporate a new sentence/document or add a word to the vocabulary.
- **Continuous Bag-of-Words (CBoW) model**: predict center word from sum of surrounding word vectors (see NLU notes)

- **Skip-gram model**: predicting surrounding single words from center word (see NLU notes). Take the target word and a neighboring context word as positive examples, randomly sample other words in the lexicon to get negative samples.
- Normalized dot-product gives **cosine similarity**. Normalize similarities to get probabilities: $P(+ \mid t, c) = 1/(1 + e^{-sim(t,c)})$
- This means we maximise the overlap (via dot product) between a word and the context it appeared in. By transitivity, any other word with a similar context will have a large overlap with the original word. For example, jumps $\sim$ leaps because their context vectors are similar.

**FastText**

- Extension of word2vec model, treats each word as composed of character n-grams.
- Can generate embeddings for OOV words, generate better embeddings than word2vec for rare words.
- Requires bigger memory and longer training times.

### 3.6 Semantic Relationship

Combining embeddings with prior knowledge: from analogical reasoning, abstract relationships were translations in the embedded space. Take this idea and extend the concept of context to include "appears in a relationship with" alongside "appears in a sentence with" and represent these new context-relationships as arbitrary affine transformations (basically, matrices).

Enforcing similarity: define an energy function $\mathcal{E}(S, R, T)$, energy is low if S is related to T through R is true (R is often non-symmetric). An example energy function is

$$\mathcal{E}(S, R, T \mid \theta) = -\frac{\mathbf{v}_T \cdot G_R \mathbf{c}_S}{||\mathbf{v}_T|| ||G_R \mathbf{c}_S||}$$

"Off-task" data helps due to shared semantic information.

### 3.7 LDA for Topic Modeling

Given the assumption of how documents are created, LDA backtracks and tries to figure out what topics would create those documents in the first place.

Start with random initial assignments and then iteratively assign words w to topics t with probability $P(t \mid d)P(w \mid t)$. After a large number of iterative steps the algorithm reaches a steady-state.

## 4 Time Series Analysis

**See MP and CIL notes on autoencoders, NLU notes on HMMs, MP notes on RNNs, Interpretability section on attention**

Main challenges in medical time series:

- Multivariate.
- Multiple data types (categorical, numeric, images, high frequency data, etc).
- Usually conditioned in some static data (e.g., patient demographics).
- Variables with different periodicities (heart rate every minute, lab tests every 6h).
- Combination of periodic and non-periodic variables.
- Data quality changes over time.
- Best practices change over time.
- Seasonality.
- New medications and techniques are introduced over time.
- Large time series.
- Right censored data (next lecture).

**Objectives of Time Series Analysis:**

- Compact description of data.
- Interpretation.
- Forecasting.
- Control.
- Hypothesis testing.
- Simulation.

**Classical decomposition:** Trend + Seasonality + Residuals

## 4.1 Auto-regressive models

An $AR(p)$ model is a regression model with lagged values of the variable (regression from previous time points to future time points; hence auto-regressive).

Auto-regressive model are suited for stationary time series. A stationary time series is one whose statistical properties such as mean, variance, autocorrelation, etc. are all constant over time. Most statistical forecasting methods are based on the assumption that the time series can be rendered approximately stationary (i.e., "stationarized") through the use of mathematical transformations.

Linear auto-regressive models:

$$Y_t = \phi_0 + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \cdots + \phi_p Y_{t-p} + \varepsilon_t$$

Non-linear auto-regressive models:

$$y_t = F(y_{t-1}, y_{t-2}, y_{t-3}, \dots) + \varepsilon_t$$

In time series modeling, a **nonlinear autoregressive exogenous model (NARX)** is a nonlinear autoregressive model which has exogenous inputs. This means that the model relates the current value of a time series to both past values of the same series and current and past values of the driving (exogenous) series — that is, of the externally determined series that influences the series of interest.

$$y_t = F(y_{t-1}, y_{t-2}, y_{t-3}, \dots, u_t, u_{t-1}, u_{t-2}, \dots) + \varepsilon_t$$

The function F is some nonlinear function, such as a polynomial. F can be a neural network, a wavelet network, a sigmoid network and so on.

Rough **classification of autoregressive models** in the lecture:

- $AR(p)$: observed, continuous
- RNN: hidden, continuous
- Markov chain: observed, discrete
- HMM: hidden, discrete

**Markov chains:**

- Next state only depends on the current state, not the past, Markov chains have no memory.
- Markov assumption: $\forall t : P(Y_t \mid Y_1, \ldots, Y_{t-1}) = P(Y_t \mid Y_{t-1})$
- Stationarity assumption: $\forall t, y, y' : P(Y_{t+1} = y \mid Y_t = y') = P(Y_t = y \mid Y_{t-1} = y')$

**RNN challenges:**

- Convergence
- Vanishing gradients etc.
- Long-range dependencies
- Many alternative models (LSTMs etc.)

Vanishing/exploding gradient problem

- Phenomenon of gradients vanishing/exploding as you get deeper (earlier) into the network
- Results in slow training, "forgetting", oscillations, instability
- LSTM proposed aimed to solve this problem

**Unitary RNN (uRNN)** is a new architecture that learns a unitary weight matrix, with eigenvalues of absolute value exactly 1. It uses parametrisation of $W_{hh}$ to create uRNN. Compares with LSTM, traditional RNN on long memory tasks

## 4.2 Unsupervised representation learning

Why representation learning: to summarize the patient time series up till the current time into a fixed-size representation, which can be used to summary the current patient state and predict the future clinical events.

Why unsupervised: large amount of unlabeled data, higher generality as unsupervised learning is agnostic to the specifics of the prediction problems, unsupervised model parameters can be used to pretrain supervised models to make them converge faster.

# 5 Survival Analysis

**Survival analysis** is defined as a set of algorithms for analysing data where the outcome variable is the time until the occurrence of an event of interest.

The **log-rank test** compares the survival times of two or more groups. The null hypothesis for a log-rank test is that the groups have the same survival.

**Censoring**:

- Left censoring – a data point is below a certain value but it is unknown by how much.
- Interval censoring – a data point is somewhere on an interval between two values.
- Right censoring – a data point is above a certain value but it is unknown by how much.
- Type I censoring occurs if an experiment has a set number of subjects or items and stops the experiment at a predetermined time, at which point any subjects remaining are right-censored.
- Type II censoring occurs if an experiment has a set number of subjects or items and stops the experiment when a predetermined number are observed to have failed; the remaining subjects are then right-censored.
- Random (or non-informative) censoring is when each subject has a censoring time that is statistically independent of their failure time. The observed value is the minimum of the censoring and failure times; subjects whose failure time is greater than their censoring time are right-censored.

General reasons why censoring might occur:

- A subject does not experience the event before the study ends
- A person is lost to follow-up during the study period
- A person withdraws from the study

**Survival function**

- $S(t) = P(T > t) = 1 - F(t)$, where $T \geq 0$ is the response variable
- The survival function gives the probability that a subject will survive past time t
- $t \in [0, \infty]$, survival function is non-increasing, $S(0) = 1$ and $S(\infty) = 0$
- In theory, survival function is smooth.

**Lifetime distribution function and event density**

- The lifetime distribution function, conventionally denoted F, is defined as the complement of the survival function: $F(t) = P(T \leq t) = 1 - S(t)$
- If F is differentiable then the derivative, which is the density function of the lifetime distribution, is conventionally denoted f: $f(t) = F'(t) = \frac{d}{dt}F(t)$
- The function f is sometimes called the event density; it is the rate of death or failure events per unit time.

**Hazard function**

- The hazard function, $h(t)$, is the instantaneous rate at which events occur, given no previous events. (The hazard function, conventionally denoted $\lambda$ , is defined as the event rate at time t conditional on survival until time t or later (that is, $T \geq t$).)
- $$h(t) = \lim_{\Delta t \to 0} \frac{P(t < T \leq t + \Delta t \mid T > t)}{\Delta t} = \frac{f(t)}{S(t)}$$
- The cumulative hazard describes the accumulated risk up to time $t$: $H(t) = \int h(u)du$

- If we know any one of the functions S(t), H(t), or h(t), we can derive the other two functions
- $H(t) = -\log(S(t))$, $h(t) = -\partial \log(S(t))/\partial t$, $F(t) = 1 - \exp(H(t))$

In the case when no event times are censored, a non-parametric estimator of $S(t) = 1 - F(t)$, where $F(t)$ is the empirical cumulative distribution. In the case when some observations are censored, we can estimate $S(t)$ using Kaplan-Meier product-limit estimator.

**Kaplan-Meier survival estimate**

$$S(t_i) - S(t_{i-1}) = (1 - \frac{d_i}{n_i})$$

where $n_i$ the number of patients alive just before $t_i$, $d_i$ the number of events at $t_i$, $t_0 = 0$, $S(0) = 1$. The estimated probability ($S(t)$) is a step function that changes value only at the time of each event. It's also possible to compute confidence intervals for the survival probability.

### 5.1 Cox proportional hazards regression

Proportional hazards models are a class of survival models in statistics. In a proportional hazards model, the unique effect of a unit increase in a covariate is multiplicative with respect to the hazard rate. For example, taking a drug may halve one's hazard rate for a stroke occurring, or, changing the material from which a manufactured component is constructed may double its hazard rate for failure.

Kaplan-Meier curves and log-rank tests are most useful when the predictor variable is categorical (e.g., drug vs. placebo), or takes a small number of values (e.g., drug doses 0, 20, 50, and 100 mg/day) that can be treated as categorical. The log-rank test and KM curves don't work easily with quantitative predictors such as gene expression, white blood count, or age. For quantitative predictor variables, an alternative method is **Cox proportional hazards regression analysis**.

## 6 Medical Image Segmentation

**Segmentation** of an image entails the division or separation of the image into regions of similar attribute.

The pixel-grid is not a natural representation of visual scenes, it is rather just an "artifact" of a digital imaging process. It would be more natural, and presumably more efficient, to work with perceptually meaningful entities obtained from a low-level grouping process.

**Superpixels** are essentially the visually homogeneous regions of an image, that were acquired by partitioning the image into N regions, where the pixels within a region share some low-level property (color, texture etc.)

# 7 Ethics

Ethics is the systematic analysis of right and wrong actions as well as of the moral principles that govern an agent's conduct.

Consent in Western medicine and medical research are founded in the ethical principle of respect for patient autonomy.

We define risk as a probability of harm and benefit as a probability of gain. We can specify the magnitude of risks/benefits by severity, duration and reversibility.

Necessary criteria for consent:
- Adequate information
- Capacity to understand
- Voluntariness

Types of risks and benefits:
- Psychological
- Sociological
- Biomedical

3Vs of Big Data
- Variety
- Velocity
- Volume

Coercion
- Explicit coercion: Use of force or intimidation to obtain compliance
- Implicit coercion: an individual is not directly forced to do X by formal coercive rules but is compelled to conform to a social equilibrium in which not doing X creates a significant disadvantage

Respect for autonomy
- Requirement that those who are capable of deliberating personal choices should be treated with respect for their capacity of self determination
- Persons with diminished or impaired autonomy or those in dependent or vulnerable positions should be protected against harm or abuse

**Minimal risk** is the probability and magnitude of physical and psychological harm that is normally encountered in the daily lives of people in a stable society or during the performance of routine physical or psychological examinations or tests.

**Big data:** extremely large datasets of heterogeneous and differently structured data produced at high frequency that can be further processed computationally.
- Promises: early detection, prevention and explanation using real-world data
- Challenges: informed consent, causality, bias, privacy and data security

**Digital phenotyping** is a multidisciplinary field of science, defined as the "moment-by-moment quantification of the individual-level human phenotype in situ using data from personal digital devices", in particular smartphones.

**Algorithmic Bias:** Data used to teach a machine learning system may reflect implicit values or attitudes of humans involved in the data collection, selection, and use.

**Algorithmic Discrimination:** Due to algorithmic bias, machine learning systems may produce outcomes that result in unjust or prejudicial treatment of different categories of people.

# 8 Privacy Preserving Methods for ML in Healthcare

**GDPR:** A regulation in EU law on data protection and privacy for all individuals within the European Union. It also addresses the export of personal data outside the EU. Aims primarily to give control to citizens and residents over their personal data and to simplify the regulatory environment for international business by unifying the regulation within the EU.

- Data storage and data sharing infrastructures meet the principles of privacy by design and by default (Art. 25)
- Consent is explicit for the data collected and the purposes data is used for (Art. 7)
- Data subjects have the right to access their personal data (Art. 15)
- Personal data can be transferred from one processing system to and into another (Art. 20)

**Typical approaches for privacy:**

- Anonymization of the data: can be freely distributed, almost impossible without losing all information, very few interesting datasets
- De-identification and regulation of the data: contractually regulate what can be done with data, often hard for the average data scientist, often requires data transfer agreement
- Providing data for analysis in a secured platform: most extreme is submitting the code and having someone executing it, more usable would be getting access to a compute environment with many technical limitations (e.g. authorized personnel, approved researchers)
- Simulating realistic data: discussed below

Anonymization refers to irreversibly severing a data set from the identity of the data contributor in a study to prevent any future re-identification, even by the study organizers under any condition. There's no re-identification of anonymized records, because the links back to the subjects are irreversibly broken. De-identification is also a severing of a data set from the identity of the data contributor, but may include preserving identifying information which can only be re-linked by a trusted party in certain situations. There is a debate in the technology community of whether data that can be re-linked, even by a trusted party, should ever be considered de-identified.

## 8.1 Synthetic Data Generation

Why generate synthetic data?

- Data could be shared and published without privacy concerns (e.g. scientific reproducibility)
- Data can be used to augment or enrich similar datasets
- Represents an alternative approach to build predictive systems
- Can benefit medical community for use in medical training simulator

RGAN loss:

$$D_{\text{loss}}(X_n, y_n) = -\text{CE}(\text{RNN}_D(X_n), y_n)$$

$$G_{\text{loss}}(Z_n) = D_{\text{loss}}(\text{RNN}_G(Z_n), 1) = -\text{CE}(\text{RNN}_D(\text{RNN}_G(Z_n)), 1)$$

$$CE(p, q) = \frac{1}{n}\sum_{i=1}^{n} H(p_i, q_i), \text{ where } H(p, q) = -\sum_x p(x) \log q(x)$$

Evaluating synthetic datasets:

- Visual inspection by experts
- Finding clusters
- Maximum Mean Discrepancy (MMD): compare statistics of the samples
- Train on Synthetic, Test on Real (TSTR)
- Train on Real, Test on Synthetic (TRTS): no as interesting as the TSTR case as it cannot diagnose mode collapse
- Mechanical Turks when no domain knowledge is needed
- Inception score for images

**MMD** is the largest difference in expectations over functions in the unit ball of a reproducing kernel Hilbert space (RKHS):

$$\sup_f(\mathbb{E}_x[f(x)] - \mathbb{E}_y[f(y)])$$

Is the GAN just memorising the training data?

- Kolmogorov-Smirnov two-sample test to compare distributions
- Nearest neighbor on training samples
- MMD three-sample test to compare synthetic data with both training and test sets

See also the related section in MP notes for more on GANs and other generative models.

## 8.2 Differential Privacy

Differential privacy addresses the case when a trusted data curator wants to release some statistic over its data without revealing information about a particular value itself.

It is a constraint on the algorithms used to publish aggregate information about a statistical database which limits the privacy impact on individuals whose information is in the database. Roughly, an algorithm is differentially private if an observer seeing its output cannot tell if a particular individual's information was used in the computation. Differential Privacy ensures that the probability that a statistical query will produce a given result is (nearly) the same whether it's conducted on the two databases, one with some specific information and one without.

The most general mechanism is known as the Laplace mechanism, which adds Laplace noise to data so that everything an adversary receives becomes noisy and imprecise, and so it is much more difficult to breach privacy (if it is feasible at all). We can also clip per-example gradients to bound the optimizer's sensitivity to individual training points.

Challenges in DP: The more information you intend to "ask" of your database, the more noise has to be injected in order to minimize the privacy leakage. Once data has been leaked, it's gone. The total allowed leakage is often referred to as a "privacy budget", and it determines how many queries will be allowed (and how accurate the results will be). "Estimation from repeated queries" is also one of the fundamental limitations of differential privacy

# 9 Interpretability of ML Models

Goal is to understand why a model is doing certain prediction. Tradeoff is that the more complex a model is, the harder it is to understand its predictions. It is worth noting that complex models can easily encode unwanted biases.

**Sensitivity Analysis of Individual Variables**

- Simple approach to obtain a ranking of variable importances. Suppress one by one the variables of the data and train the model. Measure how much the performance of the model is degraded.
- It is a global explanation method. We examine what impact each feature has on the model's prediction.
- Possible transformations that can be done during analysis are sampling uniformly from the feature distribution, permutation of the feature values, replacing the values by mean or zero.

\+ Easy to implement and useful as a sanity check.

− Computationally expensive, does not consider combinations between variables, does not explain specific predictions but the general behaviour of the model

Multiple Kernel Learning:

- Which positions in the sequence are important for discrimination?
- What characterizes these positions?
- Which motifs at which positions are important?

## 9.1 Random Forests

**Model Averaging**

- Bagging: Fit many large trees to bootstrap-resampled versions of the training data, and classify by majority vote.
- Boosting: Fit many large or small trees to reweighted versions of the training data. Classify by weighted majority vote.
- Random Forests: Fancier version of bagging

Random forests: At each tree split, a random sample of $m$ features is drawn, and only those $m$ features are considered for splitting. Typically $m = \sqrt{d}$ or $\log_2 d$, where $d$ is the number of features. For each tree grown on a bootstrap sample, the error rate for

observations left out of the bootstrap sample is monitored. This is called the "out-of-bag" error rate. Random forests tries to improve on bagging by "de- correlating" the trees.

**Mean Decrease in Impurity (MDI)**, (also called as Gini Importance) is computed by measuring how effective the feature is at reducing uncertainty (classifiers) or variance (regressors) when creating decision trees within RFs. It is defined as the total decrease in node impurity (weighted by the probability of reaching that node (which is approximated by the proportion of samples reaching that node)) averaged over all trees of the ensemble. It is fast as it is ready once RF is finished. It is often biased as it tends to inflate the importance of continuous or high-cardinality categorical variables.

**Permutation Importance**

- Train the RF and record the OOB error (classification: accuracy, regression: $R^2$)
- Permute the column values of a single feature ($j$) and then pass all test samples back through the RF and recompute the error.
- The importance of the $j^{th}$ feature is the difference between the baseline and the drop in accuracy or $R^2$ caused by permuting the column, averaged over all trees.
- z-score: standardizing the mean permutation importance with the standard error of the mean; It has been shown that the un-normalized VI "has better statistical properties"
- Permutation importance underestimates the importance of correlated predictor variables.

## 9.2 Neural Attention

Attention Mechanisms in Neural Networks: Include a subsystem in the network which is in charge of weighting the inputs data, so that the main network is fed with a preprocessed version of the data. They are very loosely inspired on humans' visual attention mechanism.

The encoder context is a weighted sum of the hidden states of the encoder. The weight of each hidden state is computed by a NN with an output softmax layer, which takes as inputs the previous decoder hidden state itself and the previous hidden states of the encoder. This additional NN which computes the weight of each hidden state is considered an attention mechanism.

Standard Encoder-Decoder framework:

- Encoder hidden states: $h_t = f(x_t, h_{t-1})$ (most common approach is RNN)
- Context: $c = q(\{h_1, \ldots, h_{\tau_x}\})$
- Example (Sutskever et al.): $f = \text{LSTM}$, $q(\{h_1, \ldots, h_{\tau_x}\}) = h_\tau$
- Decoder conditional prob.: $p(y_t \mid \{y_1, \ldots, y_{t-1}\}, c) = g(y_{t-1}, s_t, c)$
  where $q, f, g$ nonlinear functions, $s_t$ is the hidden state of the decoder RNN. Here, the context vector $c$ is the same for $\forall y_t$.

**Encoder-Decoder with Bahdanau attention:**

- Decoder conditional prob.: $p(y_i \mid y_1, \ldots, y_{i-1}, x) = g(y_{i-1}, s_i, c_i)$
- Decoder hidden state: $s_i = f(s_{i-1}, y_{i-1}, c_i)$
- Contexts: $c_i = \sum_{j=1}^{\tau_x} \alpha_{ij} h_j$

- Annotation weights:
$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^{\tau_x} \exp(e_{ik})}$$
- Energies: $e_{ij} = a(s_{i-1}, h_j)$
where $a$ is so-called alignment model, jointly trained with all other components. Unlike existing encoder-decoder models, probability in $g$ is conditioned on a distinct context vector $c_i$ for each target word $y_i$. Probability $\alpha_{ij}$ (or energy $e_{ij}$ for that reason) reflects the importance of the annotation $h_j$ w.r.t. previous hidden state $s_{i-1}$ in deciding the next state $s_i$ and generating $y_i$.

Show, attend and tell (two mechanism for obtaining context vectors from annotation vectors):

- Hard (stochastic) attention: returns a sample from every point in time, based upon a categorical distribution (of locations) parametrized by $\alpha$
- Soft attention: takes the expectation of the context vector directly
-

# 10 Appendix

A **sequence logo** consists of a stack of letters at each position. The relative sizes of the letters indicate their frequency in the sequences. The total height of the letters depicts the information content of the position, in bits.

## 10.1 Confusion Matrix

- sensitivity, recall, hit rate, or true positive rate (**TPR**)
$$\mathrm{TPR} = \frac{\mathrm{TP}}{P} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} = 1 - \mathrm{FNR}$$
- precision, positive predictive value (**PPV**)
$$\mathrm{PPV} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} = 1 - \mathrm{FDR}$$
- miss rate, false negative rate (**FNR**)
$$\mathrm{FNR} = \frac{\mathrm{FN}}{P} = \frac{\mathrm{FN}}{\mathrm{FN} + \mathrm{TP}} = 1 - \mathrm{TPR}$$
- false discovery rate (**FDR**)
$$\mathrm{FDR} = \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TP}} = 1 - \mathrm{PPV}$$
- accuracy (**ACC**)
$$\mathrm{ACC} = \frac{\mathrm{TP} + \mathrm{TN}}{P + N} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}}$$
- specificity, selectivity or true negative rate (**TNR**)
$$\mathrm{TNR} = \frac{\mathrm{TN}}{N} = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}} = 1 - \mathrm{FPR}$$
- negative predictive value (**NPV**)
$$\mathrm{NPV} = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FN}} = 1 - \mathrm{FOR}$$
- fall-out, false positive rate (**FPR**)
$$\mathrm{FPR} = \frac{\mathrm{FP}}{N} = \frac{\mathrm{FP}}{\mathrm{FN} + \mathrm{TN}} = 1 - \mathrm{TNR}$$
- false omission rate (**FOR**)
$$\mathrm{FOR} = \frac{\mathrm{FN}}{\mathrm{FN} + \mathrm{TN}} = 1 - \mathrm{NPV}$$
- **F1 score**
$$F_1 = 2 \cdot \frac{\mathrm{PPV} \cdot \mathrm{TPR}}{\mathrm{PPV} + \mathrm{TPR}} = \frac{2\mathrm{TP}}{2\mathrm{TP} + \mathrm{FP} + \mathrm{FN}}$$

## 10.2 ROC and Precision-Recall curves

**Receiver-operator curve**

It can be more flexible to predict probabilities of an observation belonging to each class in a classification problem rather than predicting classes directly. This is required when using models where the cost of one error outweighs the cost of other types of errors.

For example, in a smog prediction system, we may be far more concerned with having low false negatives than low false positives. A false negative would mean not warning about a smog day when in fact it is a high smog day, leading to health issues in the public that are unable to take precautions. A false positive means the public would take precautionary measures when they didn't need to.

The **ROC curve** is created by plotting the true positive rate (TPR) against the false positive rate (FPR) at various threshold settings.

ROC is a probability curve and AUC represents degree or measure of separability. It tells how much model is capable of distinguishing between classes. Higher the AUC, better the model is at predicting 0s as 0s and 1s as 1s. By analogy, Higher the AUC, better the model is at distinguishing between patients with disease and no disease. [link]

Area under ROC Curve (AUROC) is robust to imbalanced classes (for example, mortality has 2% positive examples)

**Precision-recall curve**

Precision can be seen as a measure of exactness or quality, whereas recall is a measure of completeness or quantity. ROC curves are appropriate when the observations are balanced between each class, whereas precision-recall curves are appropriate for imbalanced datasets.

Key to the calculation of precision and recall is that the calculations do not make use of the true negatives. It is only concerned with the correct prediction of the minority class, class 1.

Area under Precision-Recall Curve (AUPRC) quantifies the tradeoff between sensitivity and false discovery, which is relevant in a clinical setting.

## 10.3 Logarithmic scale

One reason to use the log scale is to respond to skewness towards large values; i.e., cases in which one or a few points are much larger than the bulk of the data. log scales allow a large range to be displayed without small values being compressed down into bottom of the graph.

Another reason is to show percent change or multiplicative factors. In linear scale, even if the performance in percentage terms has been fairly constant a graph of the funds will appear to have grown most rapidly at the right hand end. With a logarithmic scale a constant percentage change is seen as a constant vertical distance so a constant growth rate is seen as

a straight line. That is often a substantial advantage. In short, a logarithmic axis linearizes compound interest and exponential growth.

A logarithmic axis is useful for plotting ratios. Ratios are intrinsically asymmetrical, but ratios are symmetrical on a log scale.