**Reading Assignment 3**

L**inear regression:** It is a statistical approach for modelling the relationship between predictor variable X and a response variable Y. It assumes there is a linear relationship between these variables and we use that to predict a quantitative input. Mathematically, we can write this linear relationship as: y = β0 + β1x+ e;

where y is the output variable (also called response, target or dependent variable). e.g. house prices.

x is the input variable (also called feature, explanatory or independent variable) e.g. size of a house in squared meters.

β0 is the intercept (the value of y when x=0).

β1 is the coefficient for x and the slope of the regression line (“the average increase in Y associated with a one-unit increase in X”).

e is the error term

**Fractal clustering :**

Incremental clustering using the fractal dimension (abbreviated as Fractal Clustering for short), is a form of grid-based clustering (where the space is divided in cells by a grid). Steps that are to be followed for fractal clustering include

1. Apply GMM and inside GMM apply k-means until we get std deviation under 3 std deviations
2. And inside it,

1. Apply k-means for sub-clusters obtained from selected clusters until the objective function is satisfied

Generally, we are look at the objective function and also look into how we break a cluster in to a reasonable amount with less than 3 std deviations. Stop when std deviation of cluster > 3 std deviations.

**Entity resolution** to combine to dataset with common columns to merge or amalgamate information.

**Latent variables models** explains complex and simple relations between severable variables and an underlying unobservable i.e. latent structure.

Formally, we have a collection X = (X1,…,Xp) of manifest variables which can be observed, and a collection Y = (Y1,…, Yq) of latent variables which are unobservable and explain relationships dependency with manifest variables. For the model to be useful, ‘q’ should be much smaller than ‘p’.

Manifest and latent variables could be of 2 types, namely metrical and categorical. Metrical can be discrete or continuous while categorical is nominal or ordinal.

|  |  |  |
| --- | --- | --- |
| Latent  Manifest | Metrical | Categorical |
| Metrical | Factor analysis | Latent trait analysis or (discrete factor analysis) |
| Categorical | Latent profile analysis | Latent class analysis |

**Estimation in latent variable models**

Usually, estimation problems can be very difficult and there are problems with uniqueness of estimates, asymptotic distribution of likelihood ratio tests, also if sample sizes are small and p is not large relatively to q.

The p manifest variables X> = (X1, . . . , Xp) are linearly related to the q latent variables Y > = (Y1, . . . , Yq) as,

X = µ + ΛY + U, 🡪 (1)

where Y and U are independent and follow multivariate normal distributions

Y ∼ Nq(0, I), U ∼ Np(0, Ψ), where Ψ is a diagonal matrix, i.e. the individual error terms Ui are assumed independent. The latent variables Yj are the factors and Λ the matrix of factor loadings.

**Linear factor analysis**

The idea of the Linear NF model is to describe the variation in X by variation in a latent Y plus noise, where the number of factors q is considerably smaller than p.

The marginal distribution of the observed X is X ∼ Np(µ, Σ), Σ = ΛΛ> + Ψ.

The factor loadings Λ cannot be determined uniquely. For example, if O is an orthogonal q × q-matrix and we let

Y˜ = OY and Λ = Λ ˜ O> we have

Λ˜Y˜ = ΛO >OY = ΛY and

thus X = µ + ΛY + U = X + µ + Λ˜Y˜ + U. Since also Y˜ ∼ Nq(0, I) and

Λ˜Λ˜> = ΛO >OΛ > = ΛΛ>,

Λ and Λ˜ specify same distribution of the observable X. Hence Λ is only identifiable modulo orthogonal equivalence.

**Maximum likelihood estimation**

Expressing Λ in terms of S and ψ is more complex. Introducing

S ∗ = Ψ−1/2SΨ −1/2 , Λ ∗ = Ψ−1/2Λ.

Then the MLE of Λ ∗ can be determined by the following two criteria: 1. The columns of Λ ∗ = (λ ∗ 1 : · · · : λ ∗ q ) are eigenvectors of the q largest eigenvalues of S ∗ . 15 2. If Γ is a diagonal matrix with Γii being the eigenvalue associated with λ ∗ i , then Γii > 1, S∗Λ ∗ = Λ∗Γ.

**Orthogonal rotation**

one that ‘partitions’ the latent variables into groups of variables that mostly depend on specific factors, known as a **varimax** rotation

And a little more uncertain rotation relaxes the demand of orthogonality and allows skew coordinate systems and other variances than 1 on the latent factors, corresponding to possible dependence among the factors. Such rotations are **oblique**.

Model Assessment and Selection

The generalized performance of a learning model relates to its prediction capability on independent test data. Assessing this performance is very important as it guides the choice of learning model and gives us the measure of quality of the ultimate model that is to be chosen.

**Model selection:** It is the process of estimating the performance of different models in order

to choose the best one.

**Model assessment:** Having chosen a final model, we estimate its prediction

error (generalization error) on new data.

Bias, Variance and Model complexity

The loss function for measuring errors between Y and ˆ f(X) is denoted by

L(Y, ˆ f(X)). Typical choices are

L(Y, ˆ f(X)) = (Y − ˆ f(X))2 squared error

|Y − ˆ f(X)| absolute error.

**Generalization error** is also known as Test error which is the prediction error over an independent test sample ErrT = E[L(Y, ˆ f(X))|T ]

where both X and Y are drawn randomly from their joint distribution

(population). Here the training set T is fixed, and test error refers to the

error for this specific training set.

Training error is the average loss over the training sample.

---

err = 1/N

training error is not a good estimate of the test error,

as seen in Figure 7.1. Training error consistently decreases with model

complexity, typically dropping to zero if we increase the model complexity

enough.

**Sum of squares of error (SSE)** within cluster. SSE value will inform the user on how close each data points are to the center.

**Silhouette score:** Silhouette score measures how similar the data point is to its own cluster compared to other clusters.

**AUC-ROC curve**

It is a performance measurement for classification problem at various threshold settings

AUC describes degree or measure of separability while ROC is probability.

Chopra paper:

The paper says that the author was able to show that the desirability of a house, is not intrinsically measurable. It says about the hidden factors like latent variables improves data prediction and their dependency on multiple factors.He took an example of housing data which affected the house price model by showing that combining the two factors of desirability and characteristics provided better output than just looking at characteristics

**Lift Analysis:** Problem is: data scientists have to deal with imbalanced classes all the time, e.g. when predicting if a user will buy something in an online shop. If only 2 out of 100 customers buy anyway, it's easy for the model to predict everyone as not buying and it still would achieve an accuracy of 98%! That's absolutely not useful, when trying to assess the model's quality.

To illustrate the idea, we'll consider a simple churn model: we want to predict if a customer of an online service will cancel its subscription or not. This is a binary classification problem: the user either cancels the subscription (*churn=1*) or keeps it (*churn=0*).

The basic idea of lift analysis is as follows:

* group data based on the predicted churn probability (value between 0.0 and 1.0). Typically, you look at deciles, so you'd have 10 groups: 0.0 - 0.1, 0.1 - 0.2, ..., 0.9 - 1.0
* calculate the true churn rate per group. That is, you count how many people in each group churned and divide this by the total number of customers per group.

This model is to estimate the likeliness of it about whether a customer will cancel its subscription. This means our predicted (churn) probability should be directly proportional to the true churn probability, i.e. a high predicted score should correlate with a high actual churn rate.

**Train/Test splitting and cross validation in python**

In machine learning and statistics, we usually split our data into two subsets: training and testing data (or sometimes as train, validate and test) to fit our model on train data, in order to make predictions on the test data.

**Overfitting:** This model will be very accurate on the training data but will probably be very not accurate on untrained or new data. Overfitting means that model we trained has trained “too well” and is now, well, fit too closely to the training dataset. This usually happens when the model is too complex (i.e. too many features/variables compared to the number of observations).

**Underfitting:** It also means the model cannot be generalized to new data. In contrast to overfitting, when a model is underfitted, it means that the model does not fit the training data and therefore misses the trends in the data.

Confusion Matrix is a performance measurement for machine learning classification where output can be two or more classes.

It shows actual and predicted value combinations.It is useful for measuring recall, precision specificity and accuracy and also AUC and ROC curve.

* **Precision**  
  precision = (TP) / (TP+FP)
* **Recall**  
  recall = (TP) / (TP+FN)

TP is the number of true positives, and FP is the number of false positives.

F-score: We use this to compare two models with low precision and high recall or vice versa. It helps to measure Recall and Precision at the same time.

F-measure: (2\*Recall\*Precision) /(Recall +Precision)