



A Silicon Valley Insider

A Gentle Introduction to Machine Learning Applications for Enterprise Finance

Serge-Paul Carrasco
July 2023

**Our journey toward artificial (generalized) intelligence
(AGI)...**

Machine learning in future conversations for financial professionals



What are the benefits of using artificial intelligence and machine learning for enterprise finance and accounting workflows?

Using artificial intelligence (AI) and machine learning (ML) for enterprise finance and accounting workflows offers several benefits that can significantly enhance efficiency, accuracy, and decision-making processes. Some of the key benefits include:

1. Automation: AI and ML can automate repetitive and time-consuming tasks, such as data entry, transaction categorization, invoice processing, and report generation. This reduces the manual workload for finance and accounting teams, allowing them to focus on more strategic and value-added activities.
2. Improved Accuracy: AI and ML algorithms can analyze large volumes of financial data with precision, minimizing human errors and ensuring data accuracy. This leads to more reliable financial reporting and decision-making.
3. Fraud Detection: AI-powered systems can detect patterns and anomalies in financial transactions, helping to identify potential fraudulent activities and unauthorized transactions in real-time.
4. Predictive Analytics: ML algorithms can analyze historical financial data to predict future trends, cash flow patterns, and financial outcomes. This enables better financial planning and risk management.
5. Cost Savings: By automating tasks and streamlining processes, AI and ML can lead to cost savings in finance and accounting operations, reducing the need for manual intervention and paper-based workflows.
6. Real-time Insights: AI-powered analytics tools can provide real-time insights into financial performance, enabling faster and more informed decision-making.
7. Compliance and Regulations: AI and ML can help enterprises stay compliant with financial regulations and standards by identifying potential compliance risks and ensuring accurate reporting.
8. Scalability: AI and ML solutions can scale to handle large and complex financial datasets, making them suitable for enterprises of all sizes.
9. Improved Cash Flow Management: AI algorithms can analyze payment patterns and customer behavior, helping to optimize cash flow management and collections.
10. Enhanced Financial Forecasting: ML models can improve the accuracy of financial forecasting, allowing businesses to plan better for the future and make data-driven decisions.

Overall, integrating AI and ML technologies into finance and accounting workflows can revolutionize how enterprises manage their financial operations, leading to increased efficiency, reduced errors, better insights, and improved decision-making.

Regenerate

What chatGPT can do for enterprise finance?

Could you give a sample code of how to predict cash flow for an ERP accounting system?

Here's a sample code in Python to predict cash flow for an ERP accounting system:

```
python
import numpy as np
import pandas as pd
from sklearn.ensemble import RandomForestRegressor

# Load the accounting data into a Pandas DataFrame
df = pd.read_csv('accounting_data.csv')

# Split the data into features and target variables
X = df.drop(['Cash_Flow'], axis=1)
y = df['Cash_Flow']

# Train a Random Forest Regressor on the data
model = RandomForestRegressor()
model.fit(X, y)

# Predict the cash flow for the next period
cash_flow_prediction = model.predict([[income, expenses, accounts_receivable]])
print("Predicted Cash Flow: ", cash_flow_prediction)
```

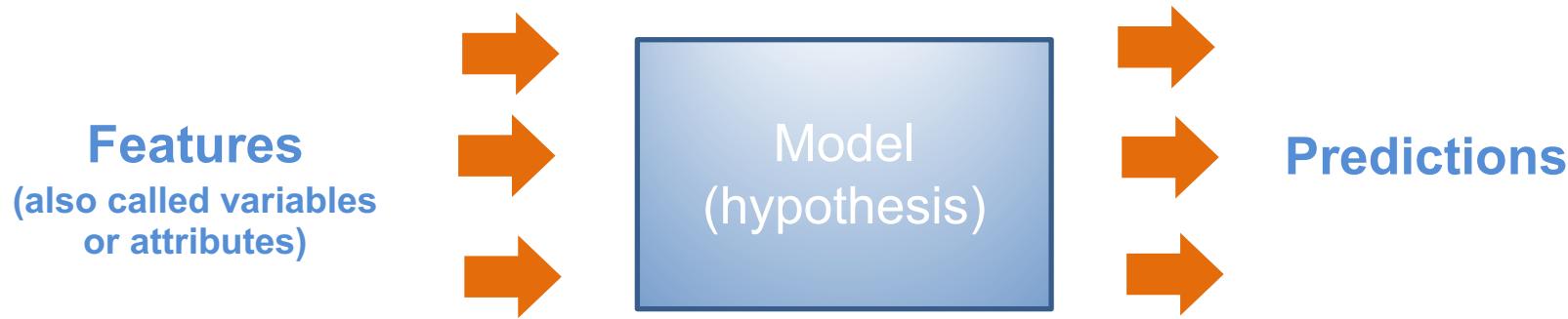
In this example, `income`, `expenses`, `accounts_receivable`, and `accounts_payable` represent the features used to predict the cash flow. The `RandomForestRegressor` is used to train the model on the accounting data in the CSV file. The model can then be used to predict the cash flow for a given set of features.

**chatGPT does not have any understanding of human language
It has been trained over very large data sets to predict co-occurrences patterns of words**

Let's be clear about what we are doing...

- **Artificial (generalized) intelligence:**
 - *“The ability of an intelligent agent to understand or learn any intellectual task that a human being can”*
- **Machine learning:**
 - *“Methods to learn from large data sets and make predictions for various tasks without any explicit instruction to follow in order to achieve that task”*
- **Deep learning:**
 - *“Deep Learning is a subfield of machine learning where the learning algorithm is based on neural networks loosely inspired by the brain cell structure”*

Model, features, and predictions



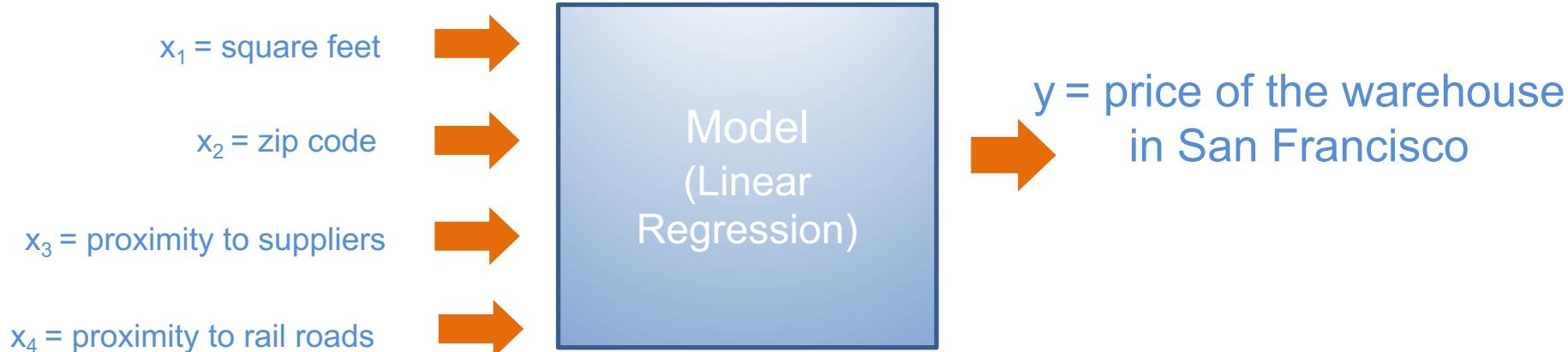
Training (or learning): the model learns how to predict an output from a training set

Inference (or prediction): once trained, the model provides its predictions with new data

Supervised learning

Predicting the price of a
warehouse

(Hello World of Machine Learning)

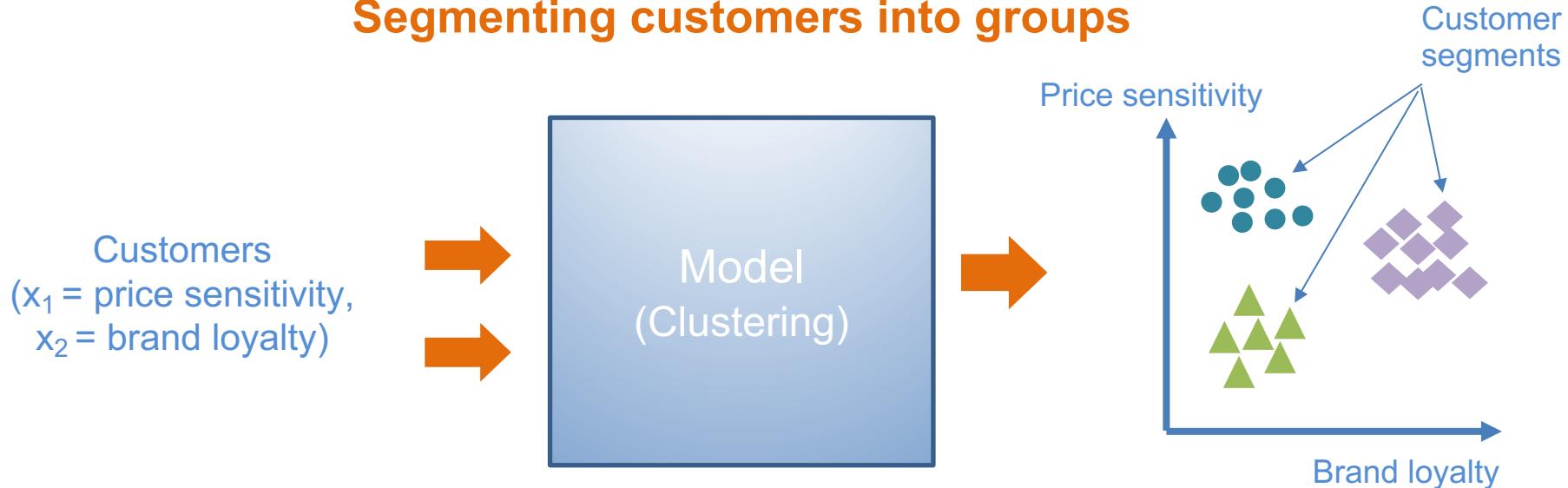


The training data are past examples of previous prices of houses

The goal of the training is to establish the “relationship” (e.g. the function and its parameters) between the prices of the warehouse and the features (square feet, zip code...)

Unsupervised learning

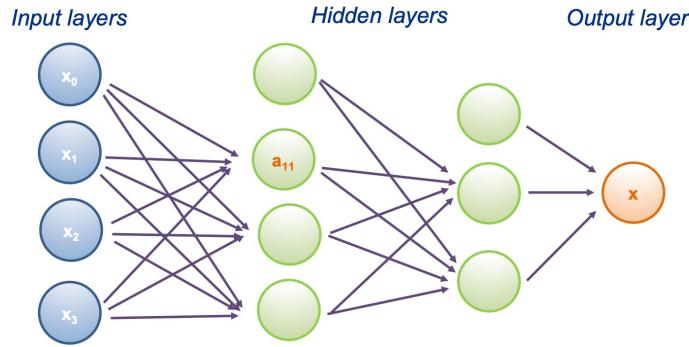
Segmenting customers into groups



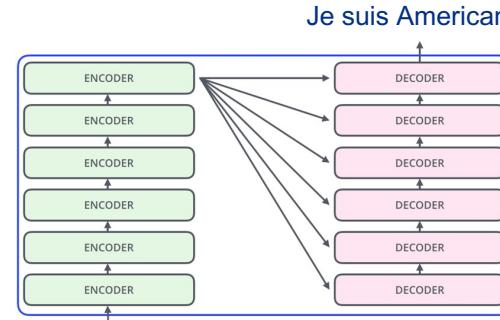
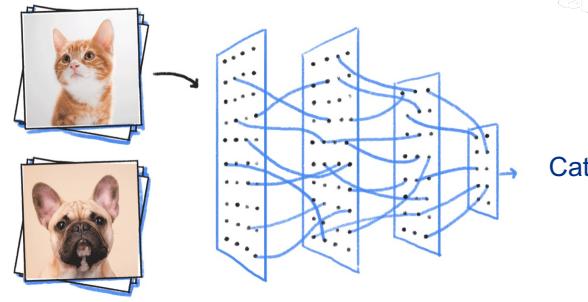
The training data set does not include the desired customer segments

The goal of the training is to provide “insights” into the data

Deep learning



**Neural networks were
“loosely” inspired by how
neurons work in the brain**



**Deep learning can learn both from
structured data and unstructured data
(text, audio, images, videos)**

Classical categorization of ML systems

Supervised Learning

Learns from examples
(called label data)

Self-Supervised Learning

Learns the examples by itself in huge amount of data

Unsupervised Learning

Learns without examples
(called unlabeled data)

Deep Learning

Learns from unstructured data (audios, images, videos...)

Reinforcement Learning

Learns from a reward

Recommender Systems

Learns from a matrix of user-item interactions

When finance meets machine learning...

What's machine learning can do for financial workflows (1)

Tasks	Use Cases & Applications	Benefits
Streamline and automate workflows	<ul style="list-style-type: none">• Increase the automation of manual, repetitive, and time-consuming tasks• At least “validate” if not “predict” data entry into accounting workflows	<ul style="list-style-type: none">• Productivity gains
Insights into day-to-day workflows	<ul style="list-style-type: none">• Detection of manual entries anomalies, internal and external fraud• Real-time insights into expenses, costs, payables, receivables to improve business operations	<ul style="list-style-type: none">• Productivity gains

What's machine learning can do for financial workflows (2)

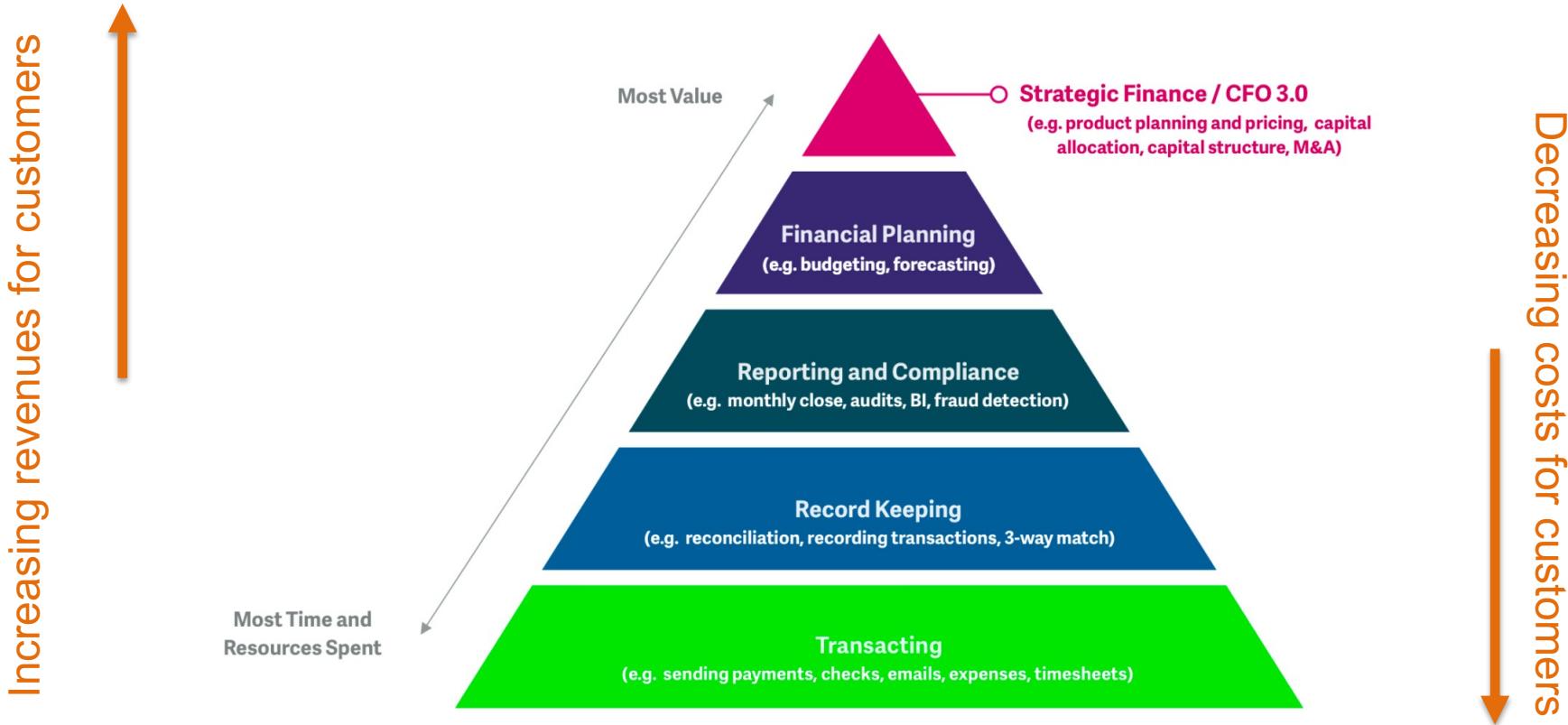
Tasks	Use Cases & Applications	Benefits
Helping with business decision making	<ul style="list-style-type: none">Budget forecasting and performance trackingRisks analyses of transactions and cash flows	<ul style="list-style-type: none">Improved revenues and profitability
Customer engagement and personalization	<ul style="list-style-type: none">Next steps and actions tailored to the customer data and workflows	<ul style="list-style-type: none">Better user experience

A sample of basic model prediction use cases for financial transactions

Examples of predictions	Models	Learning
Estimating the market value of an asset	Regression	Supervised learning
Classifying transactions	Classifier	Supervised learning
Ranking transactions	Ranking	Supervised learning
Segmenting the transactions data	Clustering	Unsupervised learning
Detecting anomalies and fraud	Outlier detection	Supervised learning
Predicting time series data	Forecasting	Supervised learning
Understanding user text and inputs	Language model	Unsupervised learning
Recommending items/action list to the user	Recommender systems	Recommender
Data extraction from a scanned document or image	Neural networks (convolutional neural networks)	Deep learning
Learning from user feedback	Neural networks (reinforcement learning)	Deep learning

Defining the machine learning models for financial workflows

Framing the opportunities to learn and automate tasks



The opportunities for core financials ML models

	Accounts Payable	Accounts Receivable	Cash Management	Purchasing
Financial Planning	Insights Prediction Forecasting	Insights Prediction Forecasting	Forecasting	Optimization Forecasting
Reporting and Compliance	Anomaly & Fraud Detection	Anomaly & Fraud Detection	Anomaly & Fraud Detection	Anomaly & Fraud Detection
Record Keeping	2-Way Match	2-Way Match	Bank Reconciliation	3-Way Match
Transacting	Extraction of bill image data			

Accounts payable

Financial Planning

- Insights into payables patterns (such as expense comparisons between departments per headcount)
- Prediction of which vendor to pay first and when (based on impacts of vendors to revenues, possible discounts to miss or not to miss)
- Forecasting of cash outflows (predictions of future bills)

Reporting and Compliance

- Anomaly detection on manual billing or validation on automatic coding/data such as: excessive/incorrect amount, duplicate billing, missing discounts, incorrect payment terms
- External fraud detection: Invalid vendor impersonating another vendor or valid vendors but abnormal transactions
- Internal fraud detection: employees who are creating un-valid bills (separation of duties)

Record Keeping

- 2-Way Match: prediction of matches between bills and purchase orders
- Prediction of account coding entries (intelligent default dimension relationships)

Transacting

- Extraction of bill image data

Accounts receivable

Financial Planning

- Insights into receivables patterns (such as possible detection of revenue leakages – similarities or differences between customer purchases)
- Prediction of when/if customer will be paying/reasons for no payments
- Forecasting of cash inflows (payments per seasonality, geography....)

Reporting and Compliance

- Anomaly detection of payment received: correct amount, correct payment terms...
- Anomaly detection of customer data
- Internal fraud detection: employees who are creating un-valid invoices (separation of duties)

Record Keeping

- 2-Way Match: prediction of matches between invoices and payments

Transacting

Cash management

Financial Planning

- Cash flow forecasting based on AP, AR, revenues, payrolls, expenses

Reporting and Compliance

- Anomaly detection on cash receipts (AR) (in particular differences in batches of receipts) and cash payments (AP)
- Fraud detection on credit card charges

Record Keeping

- Bank reconciliation: prediction of matches between bank feeds and transactions created if not create transactions
- Predictive coding on credit card charges

Transacting

Purchasing

Financial Planning

- Optimizes quantity/timing of purchases for lowest prices
- Forecasting inventories based on distributed geographies
- Forecasting inventories for cash outflows (based on 3-Way Matches)
- Forecasting required inventories for future product sales (manufacturing)

Reporting and Compliance

- Anomaly detection for purchase orders (headers, terms, entries)
- Insight into the approval process: submitters and approvers behavioral patterns
- Internal fraud detection: employees who are creating un-valid purchase orders (separation of duties)

Record Keeping

- Detection of purchase orders that could be declined based on past data
- 3-Way Matches: prediction of matches between bills, purchase orders, and good received

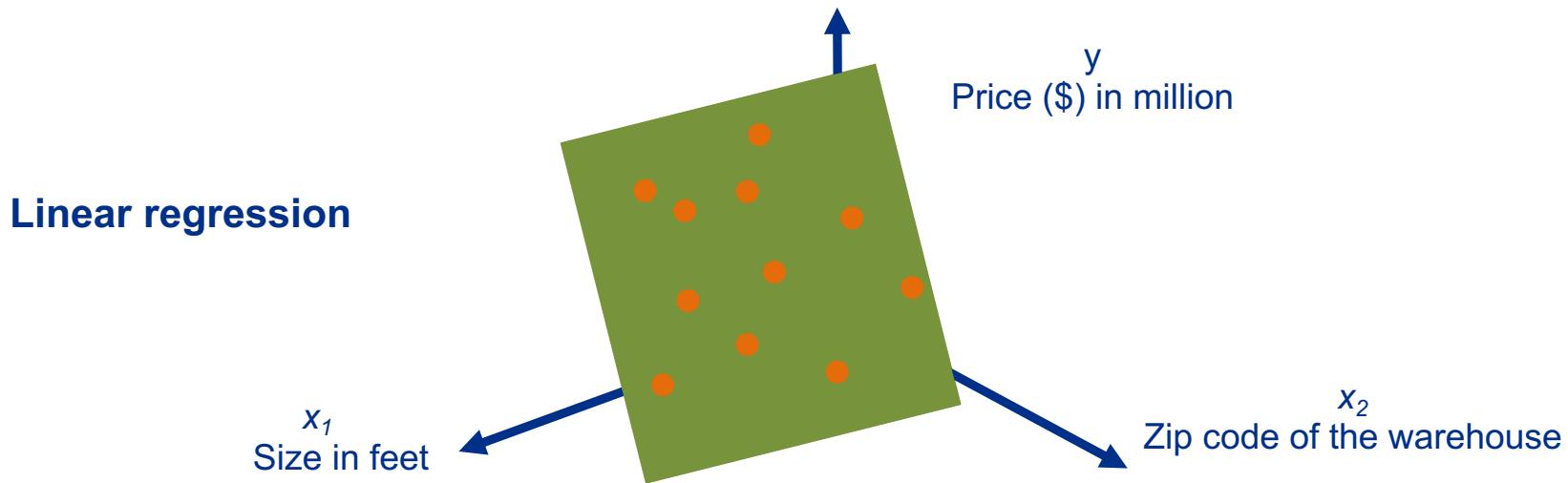
Transacting

An introduction to machine learning methods for finance

Linear Regression: Estimating a numerical value

Linear Regression
Supervised Learning

Use case: estimating the market value of an asset (a warehouse)
to raise cash

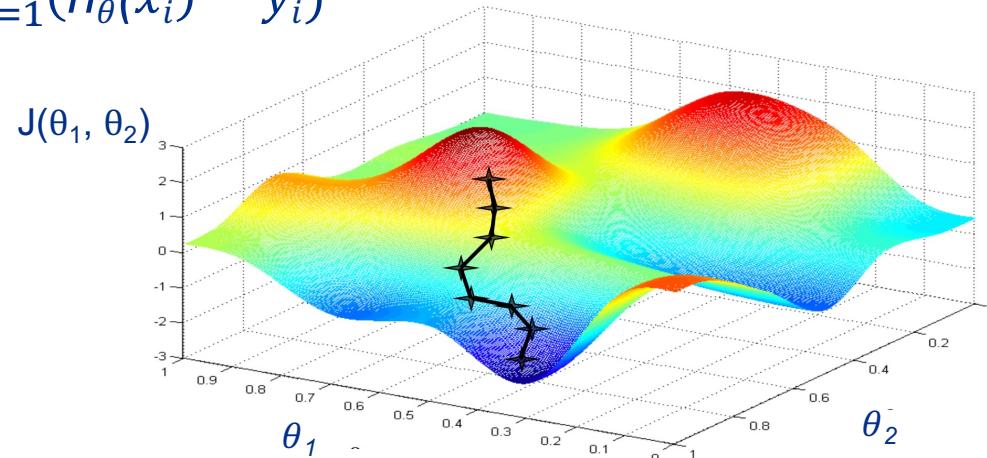
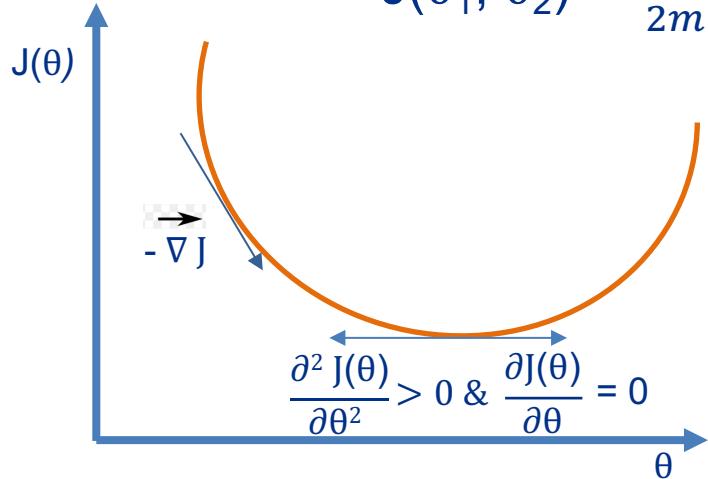


- Hypothesis: a linear regression for two variables
 - $h_{\theta}(x) = \theta_2x_2 + \theta_1x_1 + \theta_0$ (the equation of a plane)
 - The predicted price of the warehouse is a point on the plane
 - θ_2 , θ_1 and θ_0 are called the weights (or parameters) of the model

The training of a regression: Learning θ_2 , θ_1 and θ_0

We want to minimize the cost function ~ the differences between the labels and the model predictions:

$$J(\theta_1, \theta_2) = \frac{1}{2m} \sum_{i=1}^m (h_\theta(x_i) - y_i)^2$$



The negative gradient vector of the cost function $J(\theta_1, \theta_2)$ which is the vector

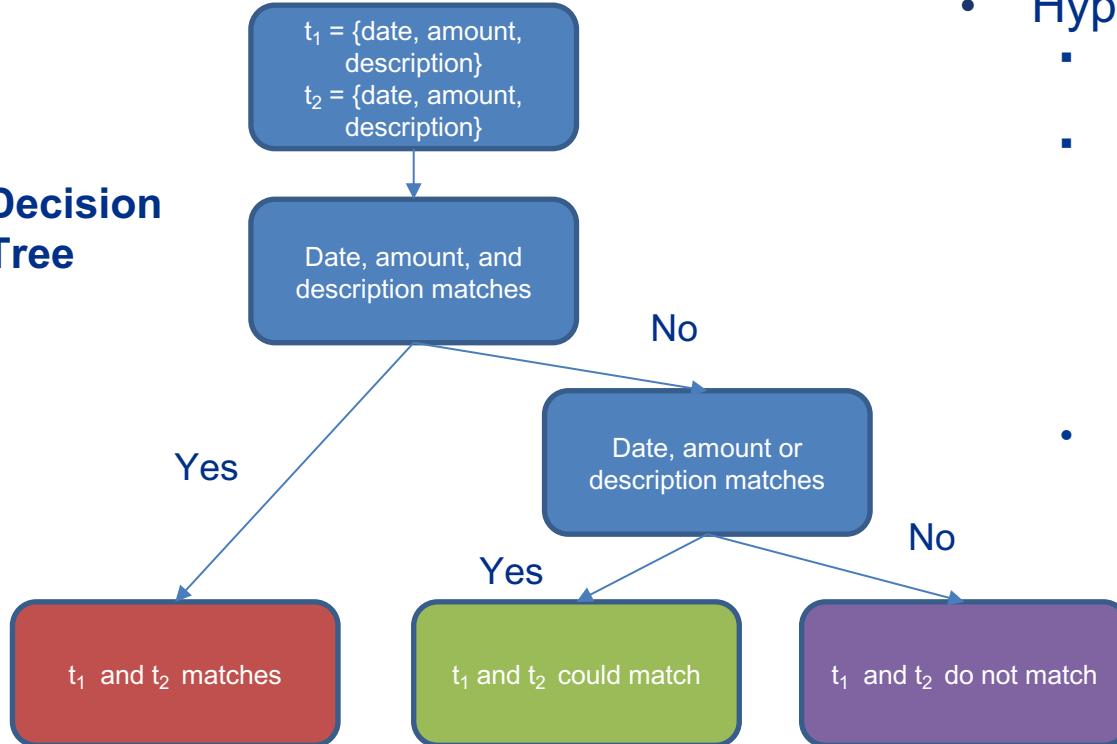
$$-\nabla J\left(\frac{\partial J(\theta_1, \theta_2)}{\partial \theta_1}, \frac{\partial J(\theta_1, \theta_2)}{\partial \theta_2}\right)$$

gives the direction to the global or a local minimum of the function ~ where the model predictions equals the labels

Classifying data into one or multiple classes

Use case: classifying transactions that can be matched

Decision Tree



- Hypothesis: a decision tree
 - How to split a node to classify the transactions?
 - The gini index measures a node impurity – a node is “pure” if all training instances, it applies to, belong to the same class:
 - $G_i = 1 - \sum_{k=1}^n p_{i,k}^2$
 - $p_{i,k}$ is the ratio of class k instances among the training instances in the i^{th} node
 - The classification and regression tree (CART) algorithm splits the training data recursively into subsets using a single feature k and a threshold t_k :
 - $J(k, t_k) = \frac{m_{left}}{m} G_{left} + \frac{m_{right}}{m} G_{right}$
 - $G \sim$ impurity of the left/right subset
 - $m \sim$ is the number of instances in the left/right subset

Random Forests: Ensemble models for decision trees

- A better way to create a stronger classification model is to aggregate the predictions of multiple decision trees in an ensemble model
- When all models are decision trees, the ensemble model is called a random forest

Random Forest

Yes

$$x = \frac{x_1 + x_2 + x_3}{3}$$



Yes
 x_1



No
 x_2



Yes
 x_3



Training data

Model performance metrics ~ How successful was the training?

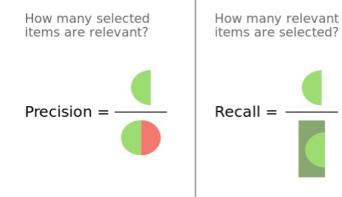
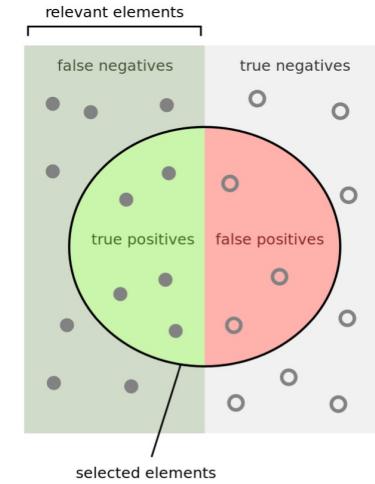
Regression

- **Root Mean Squared Error (RMSE):** Measure the distance between two vectors: the vector of the predictions and the vector of the labels

- $$\text{RMSE} = \sqrt{\frac{1}{m} \sum_{i=1}^m (h_\theta(x_i) - y_i)^2}$$
 (smaller is better)

Classification

- **Precision:** What proportion of positive identifications was actually correct?
 - $$P = \frac{\text{True positives}}{\text{True positives} + \text{false positives}}$$
- **Recall:** What proportion of actual positives was identified correctly?
 - $$R = \frac{\text{True positives}}{\text{True positives} + \text{false negatives}}$$
- **F1 score:** Accuracy of the test
 - $$F1 = \frac{2}{\frac{1}{\text{precision}} + \frac{1}{\text{recall}}} \sim 1$$
 (for a good classifier)

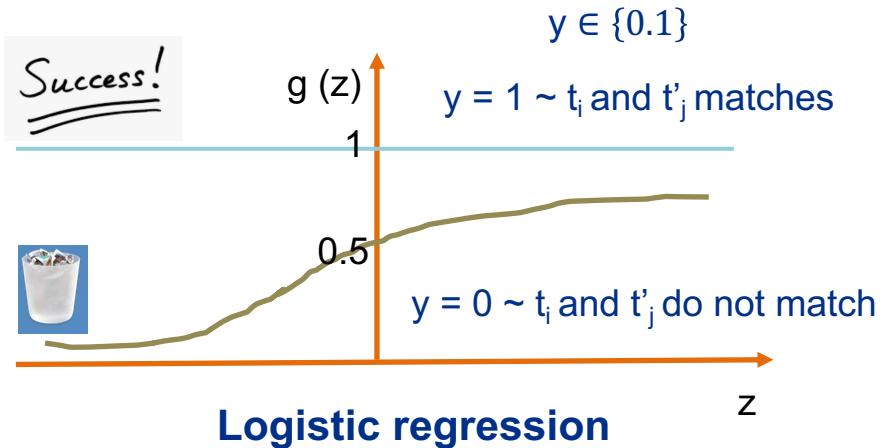


Ranking/Relevancy data sets

Logistic Regression
Supervised Learning

Use case: ranking of all transactions that belong to A and B that matches:

- $A = \{t_1, t_2, t_3, t_4, t_5 \dots\}$
- $B = \{t'_1, t'_2, t'_3, t'_4, t'_5 \dots\}$

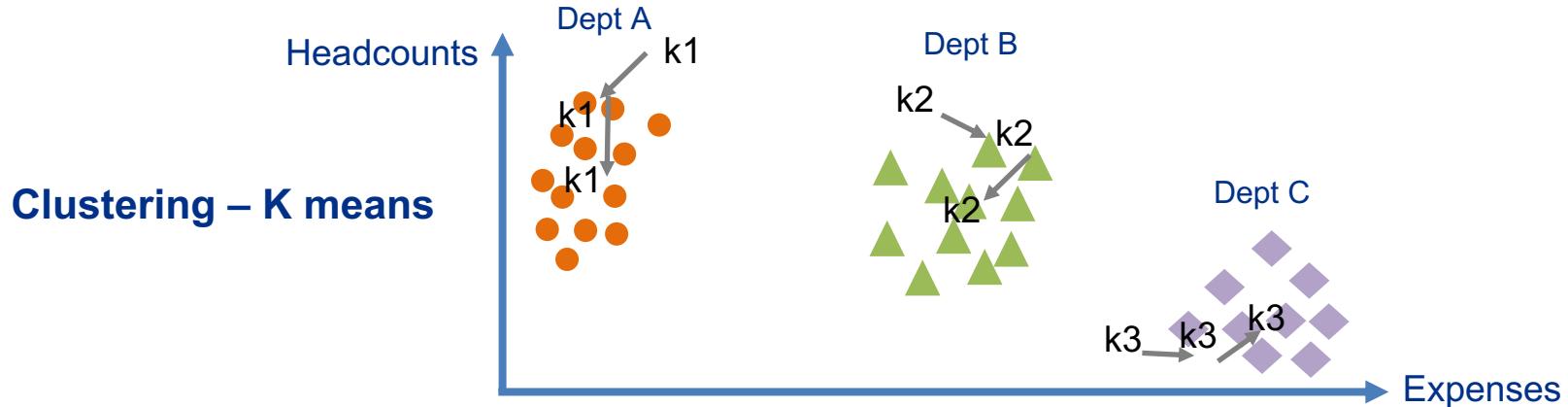


- Hypothesis: a logistic regression $h_{\theta}(x) = \frac{1}{1 + e^{-\theta^T x}}$ (a sigmoid function)
 - The sigmoid estimates a probability for each pair of transactions that matches
 - With $z = \theta^T x \Rightarrow h_{\theta}(x) = g(z) = \frac{1}{1 + e^{-z}}$
 - When $z \geq 0, g(z) \geq 0.5 \Leftrightarrow y = \text{the pair matches}$
 - When $z < 0, g(z) < 0.5 \Leftrightarrow y = \text{the pair do not match}$

Clustering: Segmenting data into groups

K-means
Unsupervised Learning

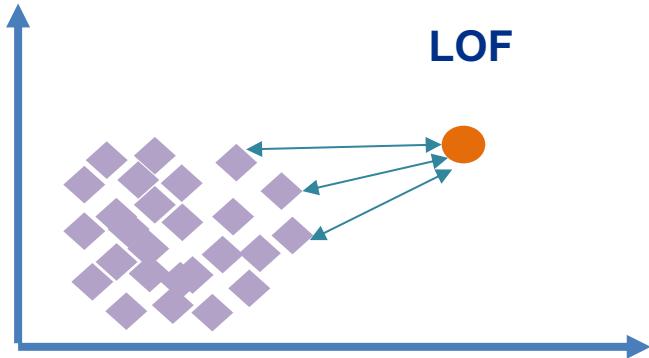
Use case: segmenting department expenses per headcounts



- Hypothesis: K-means algorithm
 - 1. Place randomly k points into the space represented by the objects that are being clustered. These points represent initial group centroids
 - 2. Assign each new observation to the group that has the closest centroid
 - 3. When all observations have been assigned, recalculate the positions of the k centroids
 - 4. Repeat steps 2 and 3 until the centroids no longer move. This produces a separation of the observations into groups

Outlier detection: Errors and frauds

Use case: detecting errors in a bill (bill header, terms of the bill, currency, bill entries)



- Hypothesis: Local outlier factor (LOF)
 - 1. The outlier is identified by the local density of its neighboring points, estimated by the distance at which a point can be "reached" from its neighbors k : $D(x_i) = \frac{1}{k} (\sum ||x_i - x_j||)$
 - 2. By comparing the local density of a point to the local densities of its neighbors, we can identify an outlier: $O(x_i) = D(x_i) / (1/k \sum D(x_j)) > 1 \Rightarrow \text{outlier}$



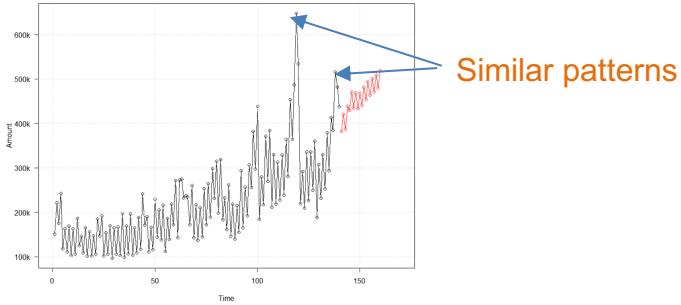
- Hypothesis: Cluster-based LOF
 - 1. Using k-means, partition data points into cluster
 - 2. For each data point, assign an outlier score based on its distance from its closest centroid, if the distance is large, the point is likely an outlier

Forecasting

Forecasting
Supervised Learning

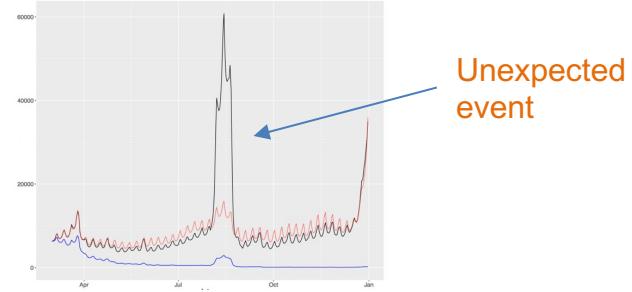
Use case: forecasting time series sales, profits, or inventory data

ARIMA



- Hypothesis: ARIMA (Autoregression Integrated Moving Average)
 - Short-term forecasting
 - **Autoregression** ~ future values are based on past values => the future will resemble to the past
 - **Integrated** ~ the time series data is made stationary ~ the statistical properties (mean, variance...) of the data is constant
 - **Moving average** ~ creates a series of averages in order to smooth out the influence of outliers

GLM



- Hypothesis: GLM models
 - Linear models:
 - $h_{\theta}(x) = \theta_2x_2 + \theta_1x_1 + \theta_0$
 - Generalized linear models (GLM):
 - $h_{\theta}(x) = \theta_2 f(x_2) + \theta_1 f(x_1) + \theta_0$
 - **Expected events ~ seasonality:**
 - GLM + seasonality = Generalized Additive Model (GAM)
 - **Unexpected events:**
 - GLM + random effects = Generalized Linear Mixed Models (GLMM)

Text processing and language model

TF-IDF and Language Models
Un/self supervised Learning

Use case: processing words in a transaction how important is a word for summarization and classification

TF-IDF

- Hypothesis: Term frequency-inverse document frequency (TF-IDF)
 - 1. Term frequency (TF) – measures how frequently a term t occurs in a document:
 - $TF(t) = (\text{Number of times term } t \text{ appears in a document}) / (\text{Total number of terms in the document})$
 - 2. Inverse-term frequency – measures how important a term is by weighing down the frequent terms while scaling up the rare ones:
 - $IDF(t) = \log (\text{Total number of documents} / \text{Number of documents with term } t \text{ in it})$
 - 3. Compute $\text{TF-IDF} = \text{TF}(t) \cdot \text{IDF}(t)$

Use case: processing a complete sentence or text from a user

Language Models

- Hypothesis: Transformer-based architecture
 - 1. Self-supervised learning ~ the model learns the labels by itself from the raw data (the text itself)
 - 2. Self-attention ~ the model understands the relationships between words
 - 3. Bi-directional ~ the model can read from right to left and left to right
 - 4. Pre-trained on many tasks: reading comprehension, summarization, translation, question answering...

Recommending items to the users

Items (dashboards, reports...)

Users

Use case: recommending dashboards, reports, tasks for an audit

- Hypothesis: Recommender systems

- **Content-based filtering:**

- Focus on the attributes of the items, and recommend to the user items based on the similarity of a given item with other items

- **Collaborative filtering:**

- Produces recommendations based on the knowledge of users' interactions with the items
- "Users who are similar to you also liked ..."
- "Users who liked this item also liked ..."

- **Latent factors:**

- Learns the latent preferences of users, and the latent attributes of items from known interactions to then predict the unknown interactions

Users

Items

User latent factors
Item latent factors

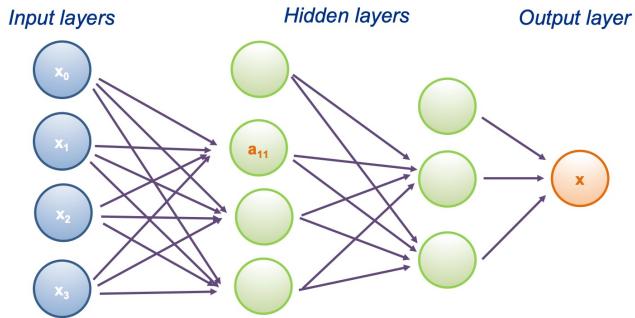
Example of latent factors: the user is a public hospital for kids in a large city in France

Deep learning: Recognizing characters and numbers

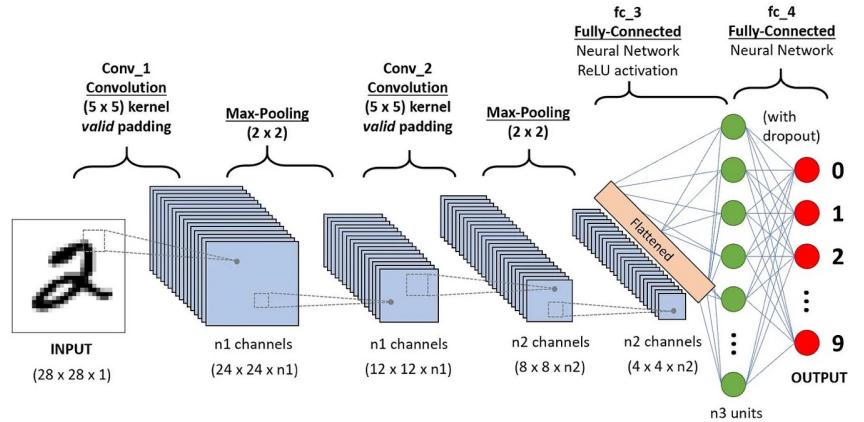
Deep Learning
Supervised Learning

Use case: Extracting data from a scanned document such as bill

Neural Network (NNs)



CNN – Computer Vision – Many NNs layers



- Hypothesis: A convolutional neural network (CNN)
 - Input to neuron 1 in layer 1: $a_{11} = \theta_3x_3 + \theta_2x_2 + \theta_1x_1$
 - Output of neuron 1 in layer 1: $h_\theta(a_{11}) - h$ is called the activation function
 - Pixels are the inputs to the CNN – the output is the digit recognized by the CNN

Deep learning: Learning from user feedback

Deep Learning
Supervised Learning

Document: xxx

Supplier: xxx

Reference: xxx

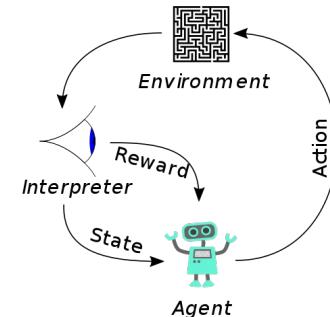
Date: xxx

Due Date: xxx

Total Amount: xxx

Transaction data is predicted, classified, anomalies are detected...an agent will improve its learning based on the user feedback

Use case: The agent collects the feedback of the user to improve the model prediction



- **Hypothesis: Reinforcement learning**

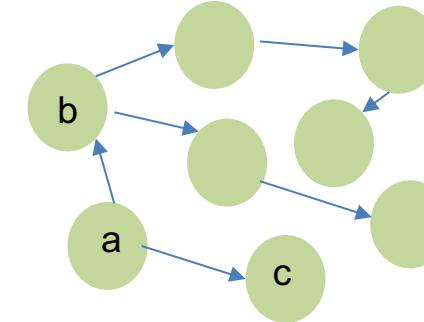
- Markov decision process (MDP). An agent interacts with the user.
- The agent chooses its next action to maximize a reward ~ the user feedback
- The agent environment changes to a new state – e.g. a more “aligned” model prediction for the user
- Balance between exploration (finding the right next move) and exploitation (maximizing future rewards)

Deep learning: Graph neural networks (GGNs) to model financial transactions

Deep Learning
Supervised Learning

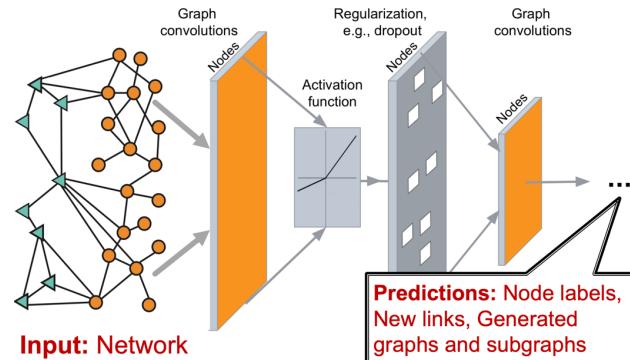
Use cases:

- Product demand forecasting, supply chain management, and optimizing inventories
- Evaluating risky financial transactions that could be illegal such as money laundering



A node can be a financial entity or a financial transaction
The edges are the relationships between the nodes

- Hypothesis: Graph neural networks
 - Predictions of node labels, new links, generated graphs and subgraphs
 - Nodes are mapped to d-dimensional encodings
 - A variant of convolution networks (CNN) is applied to the graph to enable predictions)



Combining multiple models into one workflow

From bill data extraction to generating the bill with its accounting codes

To:	Foglove Studios Limited 8 Aroha Lane, Wellington	Invoice Date	25 Jul 2022	Bayside Club Attention: Club Secretary P O Box 3354 South Mailing Center RODGE HEIGHTS MADEUPVILLE MA 60010 USA Tax reg: 90 9877899
<hr/>				
Description	Quantity	Unit Price	Tax	Amount USD
Yearly Membership Fee	1.00	230.95	8.25%	230.95
		Subtotal		230.95
		TOTAL CITY TAX 4%		9.23
		TOTAL STATE TAX 4.25%		9.82
		TOTAL USD		250.00
DUE DATE		30 Aug 2022		



Document: xxx

Supplier: xxx

Reference: xxx

Date: xxx

Due Date: xxx

Total Amount: xxx

Automatic data extraction from a scanned bill

Automatic auto-fill, coding, and anomaly and fraud detection of the bill

To matching the bill to a purchase order and reconciliating the transaction with the bank transaction

Bill

Bill: xxx
Supplier: xxx
Reference: xxx
Date: xxx
Due Date: xxx
Total Amount: xxx

Purchase Order

P.O. : xxx
Supplier: xxx
Reference: xxx
Items: xxx
Quantity: xxx
Due Date: xxx



General Ledger

Supplier 1: xxx
Supplier 2: xxx
Supplier 3 : xxx

Bank Statements

July 7th: debit xxx
July 12th : debit xxx
July 25th debit xxx

Matching the new bill to an existing purchase order

Paying the bill after validation by the user and reconciliating the bill transaction with the bank debit transaction

A short summary of the modeling approaches

Modeling approaches (1)

Coding - Bank reconciliation – 2-Way/3-Way Match

- Combination of rules and classifiers
- Rules which define the relationships between the data points
- Binary (logistic regression) and multi-class classifiers (KNN, decision trees, random forest...) to predict the relationship between those data points

Anomaly and fraud detection

- Models need to detect point, contextual or group anomalies
- Combination of many techniques that can also include rules
- Statistical profiling of data: expected range of values based on historical patterns
- Predictive confidence: classification of data based on historical patterns (logistic regression, KNN, decision trees, random forest)
- Clustering: data is clustered and outliers out of the clusters are identified (Local Outlier Detector (LOF), Cluster-based LOF (CBLOF))

Modeling approaches (2)

Predictions

- Predictive analytics based on regressions and classifiers
- Linear regressions to estimate numerical values
- Classifiers to predict an outcome or a range of possible outcomes

Insights

- Identification of patterns into data without supervision
- Clustering of data: K-means, hierarchical clustering
- Multi-class classifiers when data can be classified with supervision

Forecasting

- Time series analysis: ARIMA models
- Statistical forecasting: Generalized additive mixed models
- Causal forecasting: Finding causes-effects relationships



A Silicon Valley Insider

Thank you for your attention