Business Problems: test cases

1. Customer Churn Prediction (Prediction): At Teleco we want to predict which customers are likely to cancel their subscriptions in the next quarter so they can take proactive measures to retain them.

(https://www.kaggle.com/datasets/blastchar/telco-customer-churn)

2. Product Defect Detection (Automation): I'm a manufactorer and I want to automatically detect defective products using images captured on the assembly line, categorizing them into defect types for further inspection.

(https://www.kaggle.com/datasets/sukmaadhiwijaya/welding-defect-object-detection)

3. Employee Performance Scoring (Prediction): I want to predict employee performance scores for the upcoming evaluation period based on historical data. This will help identify high-potential employees and those who might need additional support.

Input: ratings for work-life balance, whether the employee is satisfied with their environment, self-rating.
Output: rating of the manager.

(https://www.kaggle.com/datasets/mahmoudemadabdallah/hr-analytics-employee-attrition-and-performance?select=PerformanceRating.csv)

4. Customer Segmentation (Organization): As an e-commerce company I want to segment their customers into distinct groups based on purchasing behavior and demographics to tailor marketing campaigns more effectively.

(https://www.kaggle.com/datasets/vetrirah/customer?select=Train.csv)

5. Email Classification (Automation) I want to automate the sorting of incoming customer support emails into categories such as `billing`, `technical support`, `general inquiry`, and `complaints`

(https://www.kaggle.com/datasets/tobiasbueck/multilingual-customer-support-tickets)

6. Inventory Forecasting (Prediction): As a retail company, we want to predict the demand for products in the upcoming month to optimize inventory levels and avoid stockouts or overstocking.

(https://www.kaggle.com/datasets/anirudhchauhan/retail-store-inventory-forecasting-dataset)

7. Fraud Detection in Financial Transactions (Automation): A bank wants to automatically flag potentially fraudulent transactions based on past transaction history and customer profiles. The data fields may be result of a PCA Dimensionality reduction to protect user identities and sensitive features.
(https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud)

8. Sentiment Analysis for Product Reviews (Automation): I'm an online retailer and I want to automate sentiment analysis of customer reviews to understand the overall sentiment for each product (positive, negative, or neutral).

(https://www.kaggle.com/datasets/meetnagadia/amazon-kindle-book-review-for-sentiment-analysis?select=preprocessed_kindle_review+.csv)

9. Market Basket Analysis (Organization) : A supermarket chain wants to identify frequently bought product combinations to optimize store layouts and cross-sell opportunities.
(https://www.kaggle.com/datasets/aslanahmedov/market-basket-analysis)

10. Dynamic pricing strategy (automization): A ride-sharing company wants to optimize fares in real-time based on demand fluctuations, competitor prices, and time to departure.

(https://www.kaggle.com/datasets/arashnic/dynamic-pricing-dataset)

Remarks on videos of first testing round (8 runs)

On the business analyst:

- Its guesses on the general content of supporting data are often correct.
- If the prompt is unusually specific (run 6, 7) on the data that is available, the business analyst might transfer directly to the data scientist, skipping its final response.
- There might be follow-up questions on the content of specific fields (run 3).
- It consistently gets the high-level ML solution to the business problem right.

On the datascientist:

- Often tries first to read the contents of some plausible filename, without listing the files.
- Keeps making mistakes against statefulness, e.g. in one session forgetting many times to import pandas → error rate of >50% on code blocks.
- Gets lost quite easily in directory structure. (run 6,7,8,9)
- **Not** summarizing its results, even when all code ran successfully (run 1,2,5)
- Not always getting the right target column – e.g. `SelfRating` instead of `ManagerRating` in run 2 (Employee performance).

On the deployment agent:

- It provides an ARM file, also when the data scientist fails in its task (run 6, 8), so its results are too loosely coupled to the data scientist's.

Recommendations, based on test runs:

1. Be more clear to the data scientist that it should interpret its tool results and **print a summary message** before handing off to the deployment agent; possibly with more clear examples.
2. Be more clear to the data scientist it should start with listing the files, not guessing.
3. Provide failure cases for the data scientist and the deployment agent: they should output an error condition and end the chat with TERMINATE when a file is not found, code takes too long to run, …
4. The output of the business analyst should focus more on: putting restrictions on which input to use + which output is useful.