**Movie Success Prediction Project - Complete Reference Document**

**Project Overview**

**Student:** Anton Horvat  
**Course:** AI, ML & Data | Semester 4  
**Project Goal:** Build a machine learning system that predicts movie financial success using real-world industry standards

**Success Categories:**

* **Flop:** Revenue < 1x Production Budget (immediate loss)
* **Break-even:** Revenue = 2-2.5x Production Budget (covers all costs)
* **Hit:** Revenue > 2.5x Production Budget (actual profit)

**Required Competencies**

1. **Professional Standard** - Systematic methodology, clear documentation
2. **Personal Leadership** - Independent decision-making, problem-solving
3. **Explainable AI** - Make predictions understandable to non-technical people
4. **Data Preparation & Analysis** - Thorough data work, meaningful insights
5. **Model Engineering** - Build, compare, optimize models effectively

**Current Status & Learning Foundation**

**Completed Assignments Analysis**

**Wine Dataset Assignment (Data Provisioning):**

* Systematic approach to data requirements, collection, understanding, preparation
* Used multiple visualizations: histograms, boxplots, scatterplots, correlation matrices
* Identified class separability for classification tasks
* Recognized need for feature scaling due to different value ranges

**Iris k-NN Assignment:**

* Demonstrated different results with random splits (Task 1)
* Tested feature combinations and found original 2 features sometimes better than all 4 (Task 2)
* Hyperparameter tuning showed k=5 optimal, k=85 too high (Task 3)
* Learned importance of systematic experimentation and reproducible results

**Key Insights for Movie Project**

* Feature selection requires systematic testing, not assumption that "more is better"
* Hyperparameter optimization essential for model performance
* Data scaling crucial for distance-based algorithms
* Systematic documentation and experimentation methodology needed

**Data Collection & Provisioning Framework**

**Data Collection vs Data Provisioning Relationship**

**Data Collection** = Obtaining raw data (Component within Data Provisioning)  
**Data Provisioning** = Complete process including:

1. Data Requirements (defining what you need)
2. Data Collection (actually getting the data)
3. Data Understanding (exploring and visualizing)
4. Data Preparation (cleaning and transforming)

**Planned Implementation Sequence**

1. **Phase 1:** Create TMDB API collector application (.ipynb) that extracts movie data to CSV
2. **Phase 2:** Data provisioning notebook using the CSV with 12-16 visualizations

**Data Collection Best Practices (From Lecture)**

1. **Start with Data Requirements** - Know what you're looking for
2. **Collect the Data** - APIs, databases, existing datasets, web scraping
3. **Storage Strategy** - Consider 24/7 availability, versioning, naming systems
4. **Retrieval Planning** - How often to reload, handling updates
5. **Integration/Limitation** - Extend or limit dataset as needed
6. **Documentation** - Create detailed data dictionary

**Data Integration Techniques Available**

1. **Union** - Sum of both datasets (same structure required)
2. **Inner Join** - Intersection of datasets
3. **Left/Right Outer Join** - One dataset + matching records (creates NULL values)
4. **Full Outer Join** - Combination of left and right outer joins
5. **Exclusion** - Reduce datasets by eliminating unwanted characteristics

**Join Tips:**

* Always join on dimension attributes
* Use same grain
* Join attribute is lowest level of hierarchy
* Use algorithms for geolocation vicinity joining

**Data Dictionary Requirements**

For each data element, document:

* **Data Element Name** - Descriptive, clear name
* **Data Type** - Numerical, categorical, text, image, time series
* **Units** - Measurement units for numerical data
* **Range** - Valid values or ranges
* **Description** - What the element represents
* **Source** - Where data is obtained (database, API, sensor)
* **Quality Standards** - Accuracy, completeness, formatting requirements
* **Notes** - Additional context or comments

**Movie Project Data Strategy**

**TMDB API Collection Plan**

**Primary Data Source:** TMDB (The Movie Database) API **Target Output:** CSV file with comprehensive movie features

**Required Features for Success Prediction:**

* **Financial:** Budget, Revenue (for target variable creation)
* **Content:** Genre, Runtime, Release Date, Rating
* **People:** Director, Main Cast, Production Company
* **Performance:** User Ratings, Vote Counts

**Target Variable Creation:**

python

def classify\_success(revenue, budget):

if budget == 0 or revenue == 0:

return None

ratio = revenue / budget

if ratio < 1.0:

return "Flop"

elif ratio < 2.5:

return "Break-even"

else:

return "Hit"

**Data Provisioning Visualization Plan (12-16 visualizations)**

Based on wine assignment methodology:

**Distribution Analysis:**

1. Revenue distribution (histogram with different bin sizes)
2. Budget distribution (log-transformed)
3. Success category distribution (bar chart)
4. Genre distribution (bar chart)

**Relationship Analysis:** 5. Budget vs Revenue scatter plot (colored by success) 6. Genre vs Success rate (boxplot) 7. Release month vs Success (boxplot) 8. Runtime vs Success correlation

**Advanced Visualizations:** 9. Director success rate analysis 10. Production company performance 11. Correlation matrix (all numerical features) 12. Feature importance from preliminary models

**Temporal Analysis:** 13. Success trends over years 14. Seasonal release patterns 15. Budget inflation over time

**Comparative Analysis:** 16. Genre-specific budget vs revenue relationships

**Technical Implementation Notes**

**Environment:** Python + Jupyter Notebooks  
**Data Source:** TMDB API + OMDb API  
**Key Libraries:** pandas, matplotlib, seaborn, scikit-learn

**Code Documentation Style:**

* Write explanatory text above each code block explaining purpose and benefit to AI model
* Avoid excessive inline comments
* Focus on methodology explanation rather than syntax explanation

**Learning Integration:**

* Reference wine assignment learnings for data provisioning structure
* Apply iris assignment insights for systematic model experimentation
* Document all decisions and rationale for reproducibility

This reference document contains all the information from your assignments, lecture materials, and project requirements. You can use this for our next chat when we start building the TMDB API collector and subsequent data provisioning notebook for your movie success prediction project.