
SkyGeni – Sales Intelligence Challenge

Role: Data Science / Applied AI Engineer

Time Expectation: 6–8 hours

Submission Format: GitHub repository + README

Background

SkyGeni builds AI-driven sales intelligence systems for B2B SaaS companies. Our customers are typically CROs, Sales Leaders, and Revenue Operations teams who want to understand:

- Why deals are being won or lost
- What risks exist in the current pipeline
- Which actions can improve revenue outcomes

You are joining SkyGeni as an early Data Science / Applied AI Engineer. Your role is not just to build models, but to design **decision intelligence systems** that help business leaders take better actions.

This assignment simulates a real problem you might face at SkyGeni.

Business Scenario

A B2B SaaS company uses SkyGeni to monitor their sales performance.

The CRO complains:

“Our win rate has dropped over the last two quarters, but pipeline volume looks healthy.

I don’t know what exactly is going wrong or what my team should focus on.”

You are asked to:

Investigate the problem and design a data-driven insight system that can help the CRO understand what is happening and what actions to take.

Provided Data

You will be given (or you can generate) a dataset with the following typical fields:

- deal_id
 - created_date
 - closed_date
 - deal_stage
 - deal_amount (ACV)
 - sales_rep_id
 - industry
 - region
 - product_type
 - lead_source
 - outcome (won / lost)
-

Your Tasks

This is intentionally an **open-ended problem**. There is no single correct solution.

We care more about:

- How you think
- How you frame the problem
- How you reason about business decisions

Not about fancy ML models.

Part 1 – Problem Framing (No Code Required)

Write a short section answering:

1. What do you think is the **real business problem** here?
 2. What key questions should an AI system answer for the CRO?
 3. What metrics matter most for diagnosing win rate issues?
 4. What assumptions are you making about the data or business?
-

Part 2 – Data Exploration & Insights

Using Python:

1. Perform exploratory data analysis (EDA)
 2. Identify at least:
 - **3 meaningful business insights**
 - **2 custom metrics** you invent yourself (not just standard ones)
 3. Explain each insight in plain business language:
 - Why does it matter?
 - What action could it drive?
-

Part 3 – Build a Decision Engine

Choose **ONE** of the following problems (your choice):

Option A – Deal Risk Scoring

Score open deals by probability of loss.

Option B – Win Rate Driver Analysis

Identify which factors are hurting or improving win rate.

Option C – Revenue Forecast

Predict expected revenue for the next period.

Option D – Pipeline Anomaly Detection

Detect unusual changes in pipeline behavior.

For your chosen option:

- Define the problem clearly
- Build a simple model or rule-based system
- Generate **actionable outputs**
- Explain how a sales leader would use this

We care about:

- Reasonable modeling choices

- Interpretation
 - Business usefulness
- Not Kaggle-style accuracy.
-

Part 4 – Mini System Design

Design a lightweight **Sales Insight & Alert System**.

Include:

- High-level architecture (diagram optional)
- Data flow
- Example alerts or insights
- How often it runs
- Failure cases and limitations

Think like:

“If SkyGeni were to productize this, what would it look like?”

Part 5 – Reflection (Most Important)

Write a short reflection:

1. What assumptions in your solution are weakest?
 2. What would break in real-world production?
 3. What would you build next if given 1 month?
 4. What part of your solution are you least confident about?
-

Submission Instructions

Please submit:

1. GitHub repository containing:
 - All code
 - README.md explaining:
 - Your approach

- How to run the project
 - Key decisions
 - 2. Optional:
 - Short Loom / video (max 5 mins) explaining your solution
-

Evaluation Criteria

You will be evaluated on:

Area	Weight
Problem framing & business thinking	25%
Insight quality & metric design	20%
Decision engine quality	20%
Engineering & code quality	15%
Out-of-box thinking	10%
Communication & clarity	10%

Total: **100%**

Important Notes

- You are free to use:
 - Google
 - StackOverflow
 - GitHub Copilot / ChatGPT
- We assume modern engineers use tools.

What we evaluate is:

Your thinking, decisions, and judgment — not your ability to write syntax from memory.

There is no single correct solution.
We are interested in *how you think* more than *what you build*.
