LINEAR REGRESSION MODEL FOR PREDICTIVE SERVICE PROVIDER SELECTION

Tamerlan Aghayev

Salviano Diamond

CCAAW 2023

Outline

- Problem Statement
- Methodology
- Demonstration
- Future Work
- Conclusion
- Q & A

Problem Statement

- **Aging NASA satellite constellation:** Need for updated, flexible communication systems

- **Manual intervention limitations:** Current methods, involving human-led negotiations with service providers, are increasingly impractical

- **Requirement for automation:** An automated system for resource scheduling and data management is necessary

- **Recommender system development:** Proposal for a new system that efficiently recommends top-ranked providers based on prior performance

Methodology

I. **Model data**
- Define provider, spacecraft, and user requirements
- Preprocess the data
- Cache the data into comma separated values (CSV)

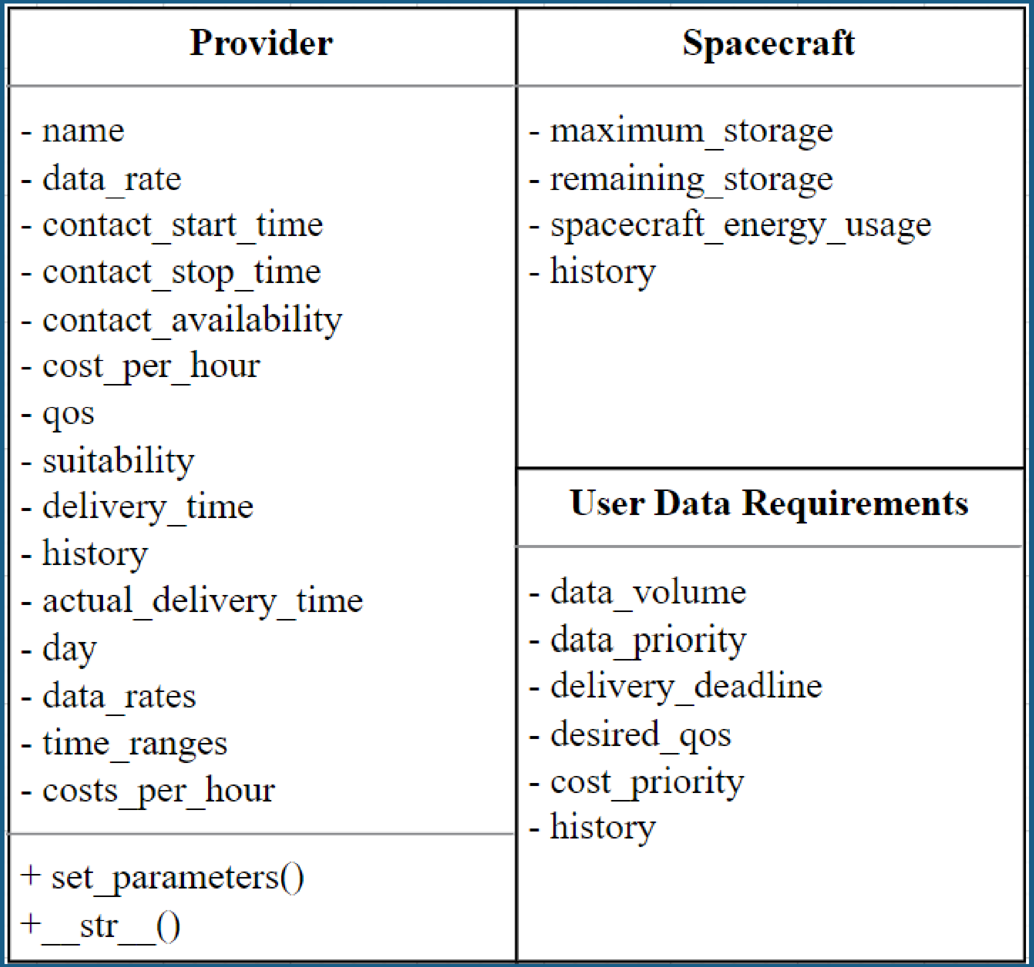
II. **Evaluate Providers**
- Process generated/user input data to determine available providers
- Evaluate available providers, append to CSV files for each provider
- Determine list of available providers sorted by scalar quantity - suitability

III. **Predict rankings**
- Ingest all CSV files from the Link Selection Algorithm (LSA)
- Process all rows and columns to validate data
- Generate linear regression models to predict provider’s future suitability value

I. Model Data

Model Data

- Object - oriented design
  - Provider
  - Spacecraft
  - User Data Requirements
- Providers “change in time”
- Data is stored locally



Provider

- name
- data_rate
- contact_start_time
- contact_stop_time
- contact_availability
- cost_per_hour
- qos
- suitability
- delivery_time
- history
- actual_delivery_time
- day
- data_rates
- time_ranges
- costs_per_hour

+ set_parameters()
+ __str__()

Spacecraft

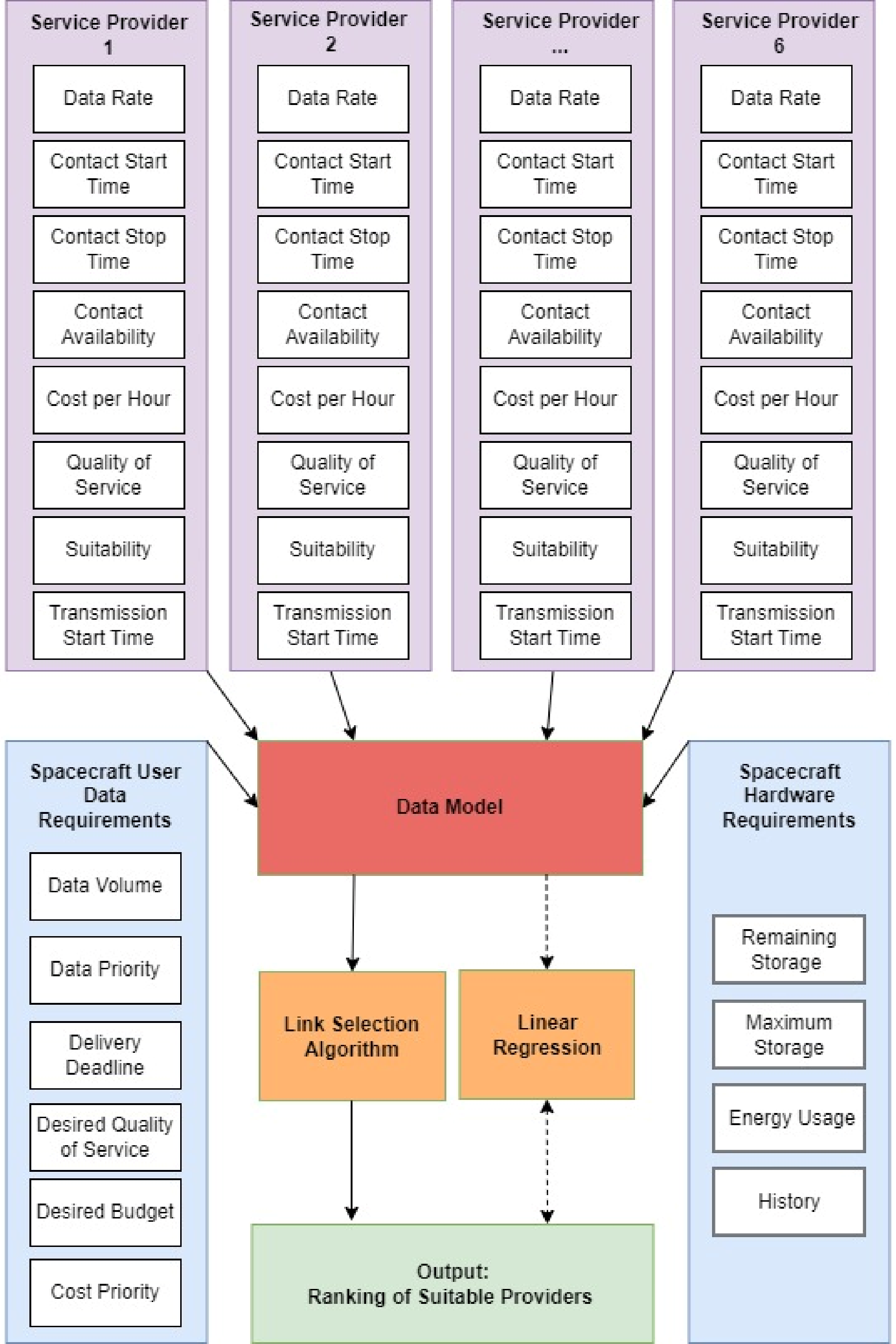
- maximum_storage
- remaining_storage
- spacecraft_energy_usage
- history

User Data Requirements

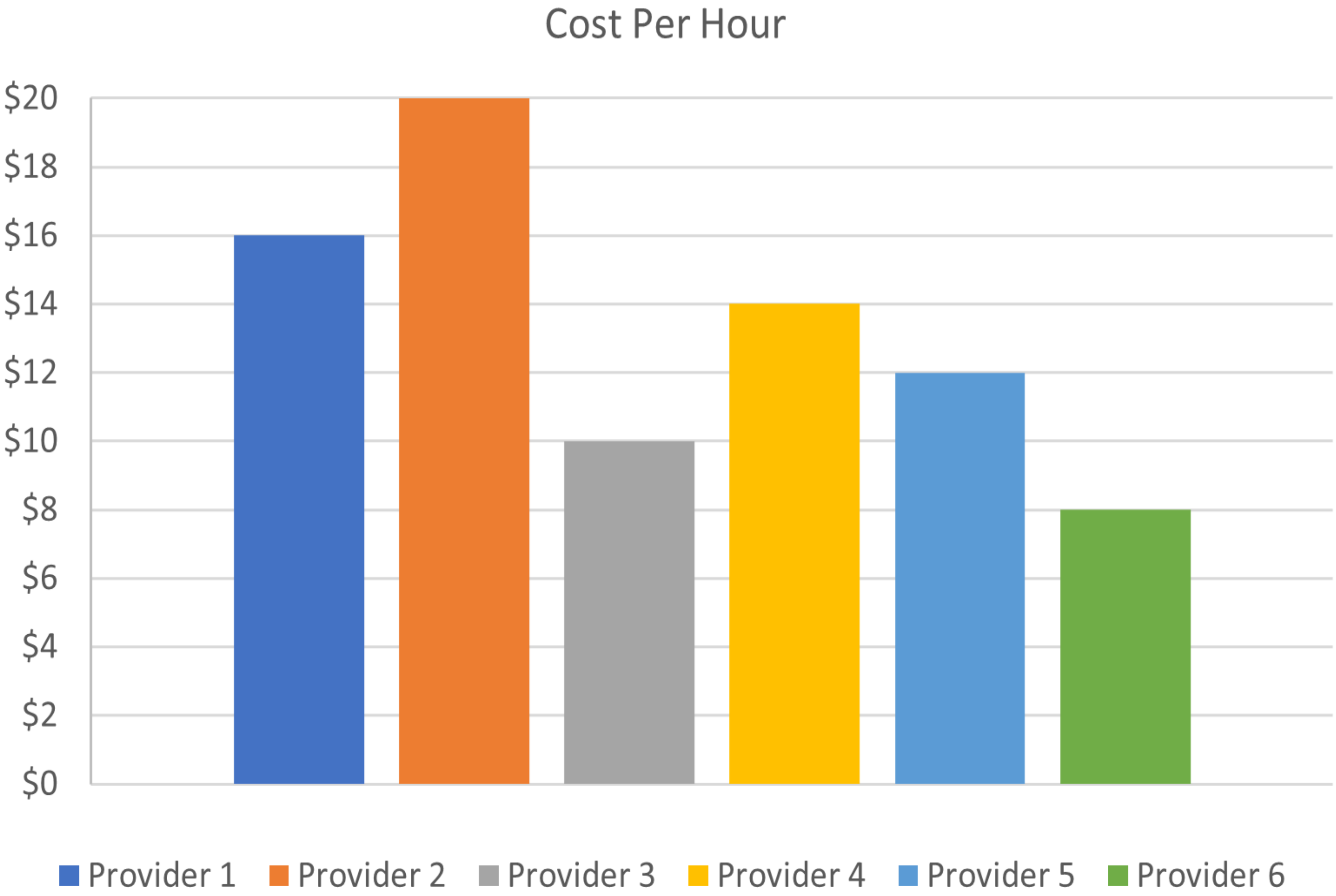
- data_volume
- data_priority
- delivery_deadline
- desired_qos
- cost_priority
- history

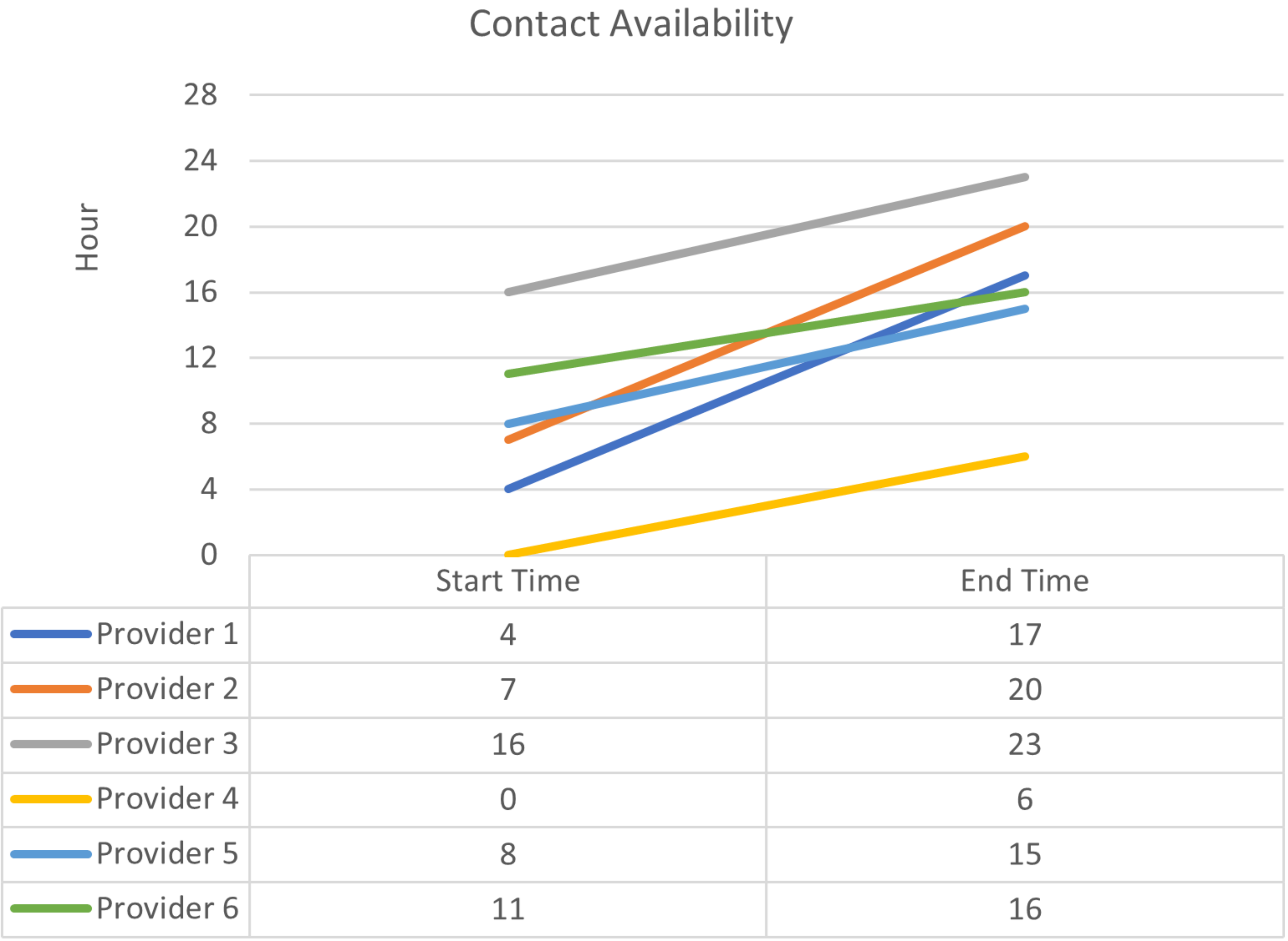
Data Objects

Data Model and Link Selection Decision Process



Data values are notional only and do not represent actual costs or data rates


Cost Per Hour
$20
$18
$16
$14
$12
$10
$8
$6
$4
$2
$0
Provider 1
Provider 2
Provider 3
Provider 4
Provider 5
Provider 6

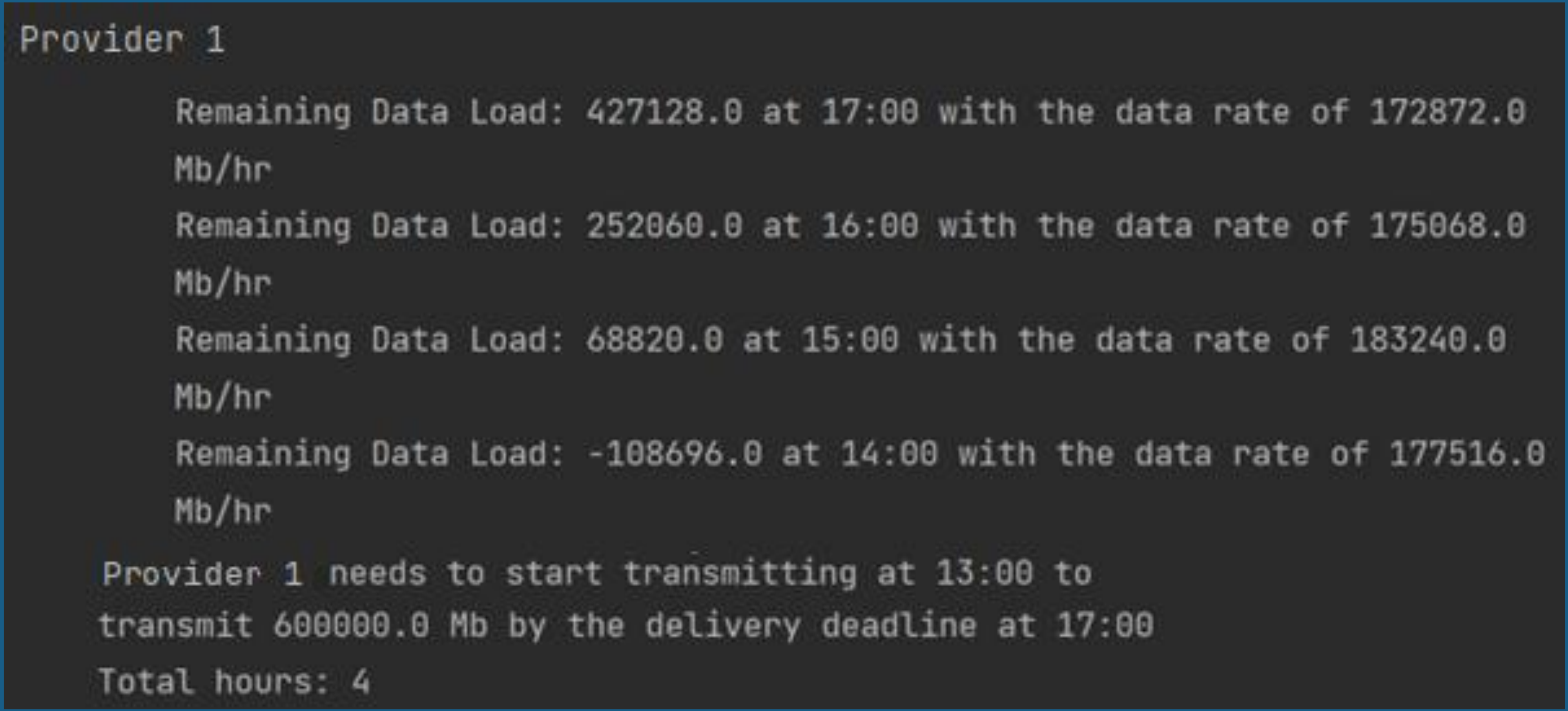


	Start Time	End Time
Provider 1	4	17
Provider 2	7	20
Provider 3	16	23
Provider 4	0	6
Provider 5	8	15
Provider 6	11	16

II. Evaluate Providers

Link Selection Algorithm - Availability

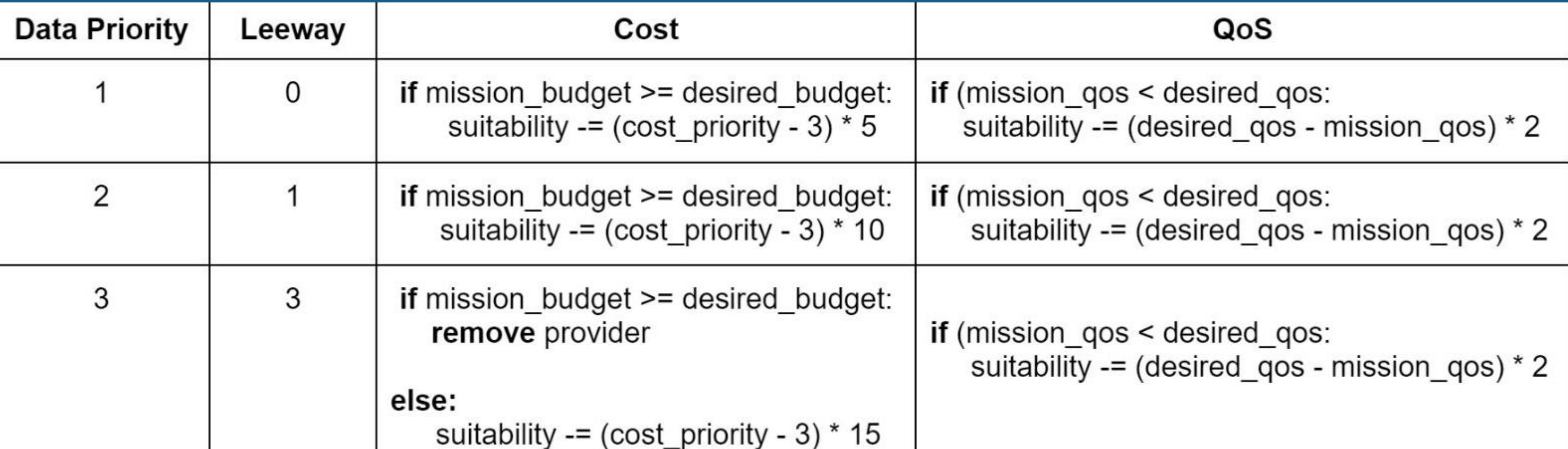
Determine which providers are available at mission time,
and can complete transmission



```
Provider 1
        Remaining Data Load: 427128.0 at 17:00 with the data rate of 172872.0
        Mb/hr
        Remaining Data Load: 252060.0 at 16:00 with the data rate of 175068.0
        Mb/hr
        Remaining Data Load: 68820.0 at 15:00 with the data rate of 183240.0
        Mb/hr
        Remaining Data Load: -108696.0 at 14:00 with the data rate of 177516.0
        Mb/hr
    Provider 1 needs to start transmitting at 13:00 to
    transmit 600000.0 Mb by the delivery deadline at 17:00
    Total hours: 4
```

Link Selection Algorithm - Suitability

Leeway - the amount of time in hours past the delivery deadline
Suitability – grade that evaluates how well a provider fits a given mission



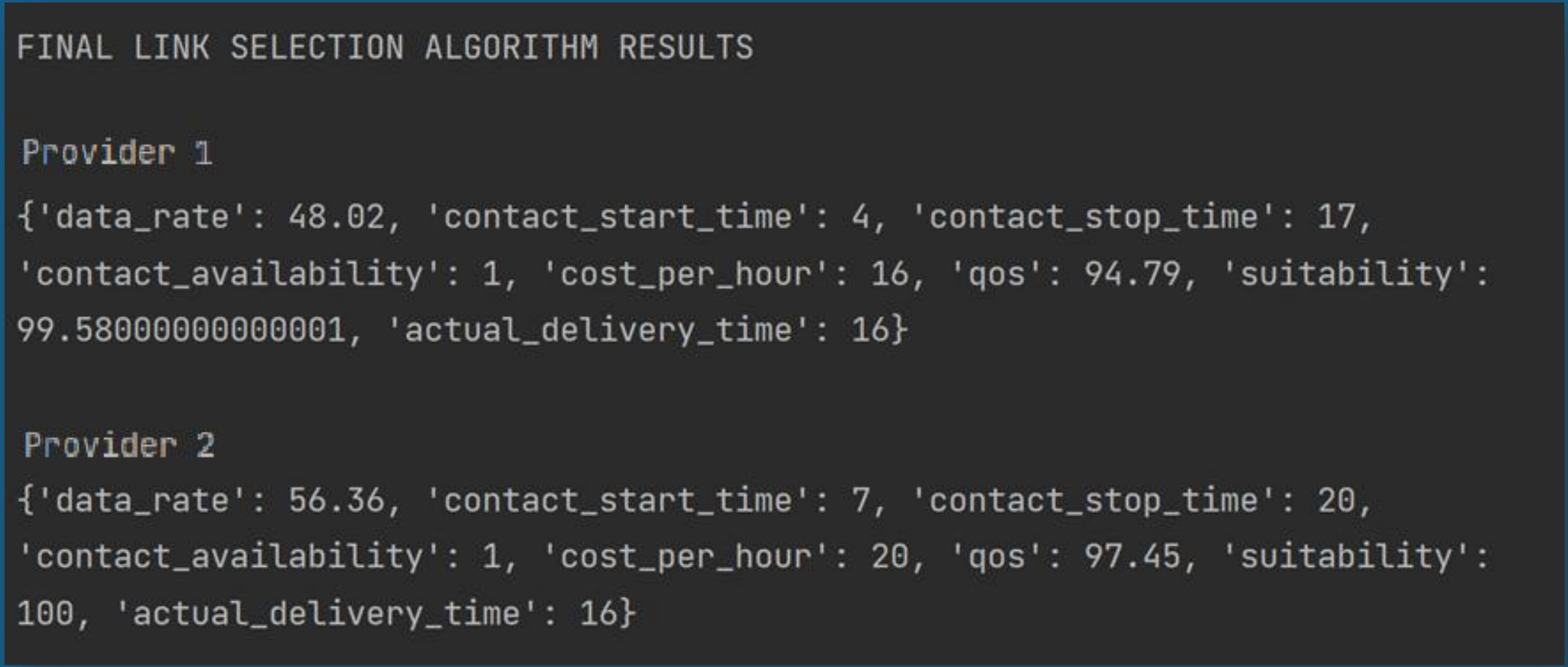
Data Priority	Leeway	Cost	QoS

1	0	**if** mission_budget >= desired_budget:
		suitability -= (cost_priority - 3) * 5
			if (mission_qos < desired_qos:
			suitability -= (desired_qos - mission_qos) * 2

2	1	**if** mission_budget >= desired_budget:
		suitability -= (cost_priority - 3) * 10
			if (mission_qos < desired_qos:
			suitability -= (desired_qos - mission_qos) * 2

3	3	**if** mission_budget >= desired_budget:
		remove provider
		else:
		suitability -= (cost_priority - 3) * 15
			if (mission_qos < desired_qos:
			suitability -= (desired_qos - mission_qos) * 2

Link Selection Algorithm - Output



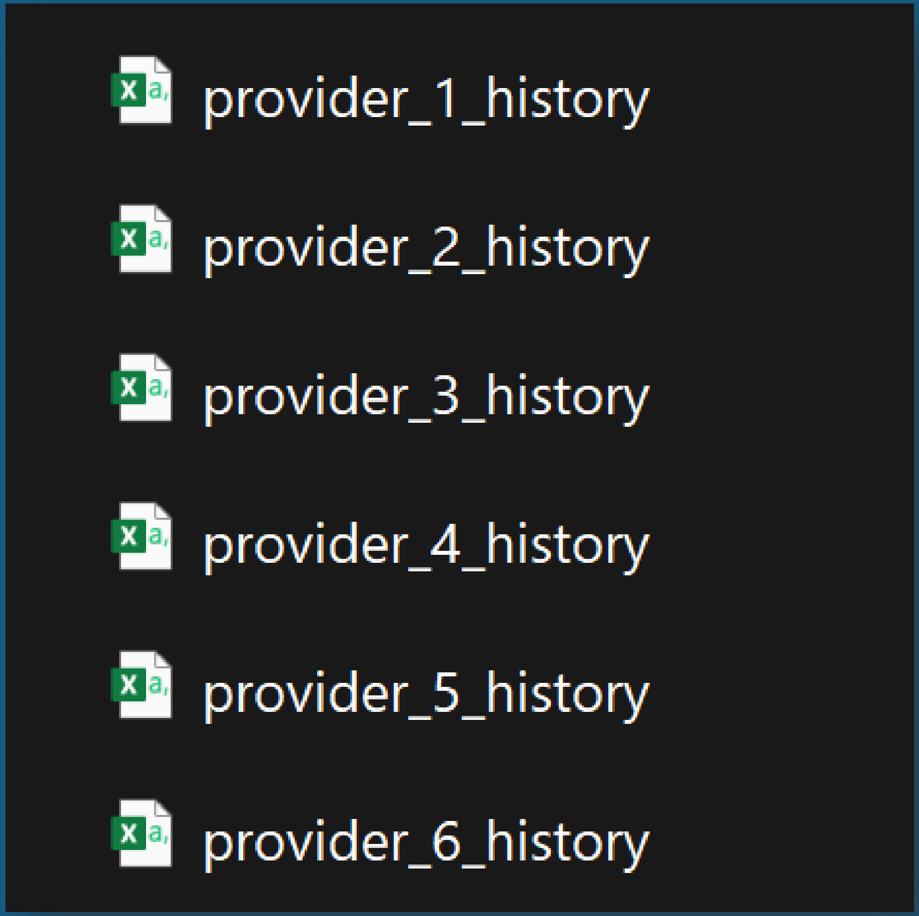
```
FINAL LINK SELECTION ALGORITHM RESULTS

Provider 1
{'data_rate': 48.02, 'contact_start_time': 4, 'contact_stop_time': 17,
'contact_availability': 1, 'cost_per_hour': 16, 'qos': 94.79, 'suitability':
99.58000000000001, 'actual_delivery_time': 16}

Provider 2
{'data_rate': 56.36, 'contact_start_time': 7, 'contact_stop_time': 20,
'contact_availability': 1, 'cost_per_hour': 20, 'qos': 97.45, 'suitability':
100, 'actual_delivery_time': 16}
```

III. Predict Rankings

LSA
+
Linear Regression

- Each provider has a DataFrame generated, based on cached CSV files (from LSA)

- If provider’s DataFrame contains sufficient link selections at delivery deadline



provider_1_history

provider_2_history

provider_3_history

provider_4_history

provider_5_history

provider_6_history

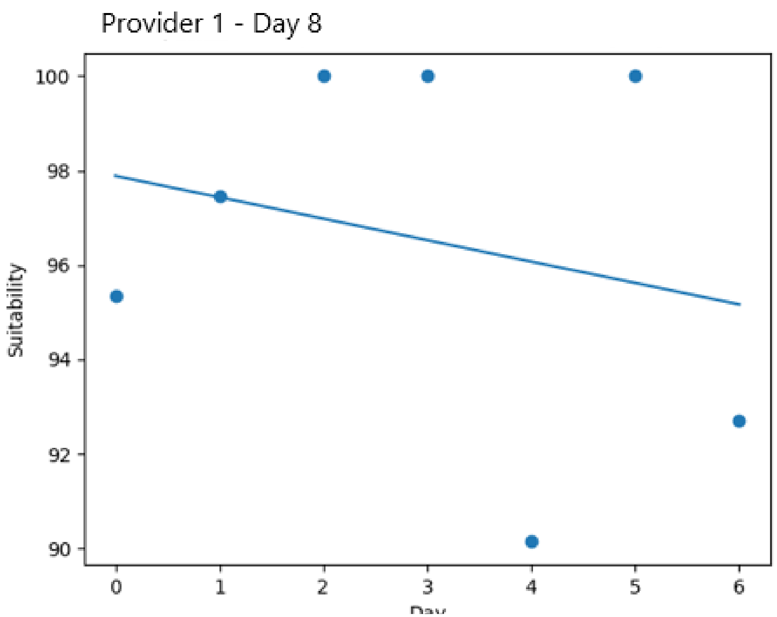
Linear Regression

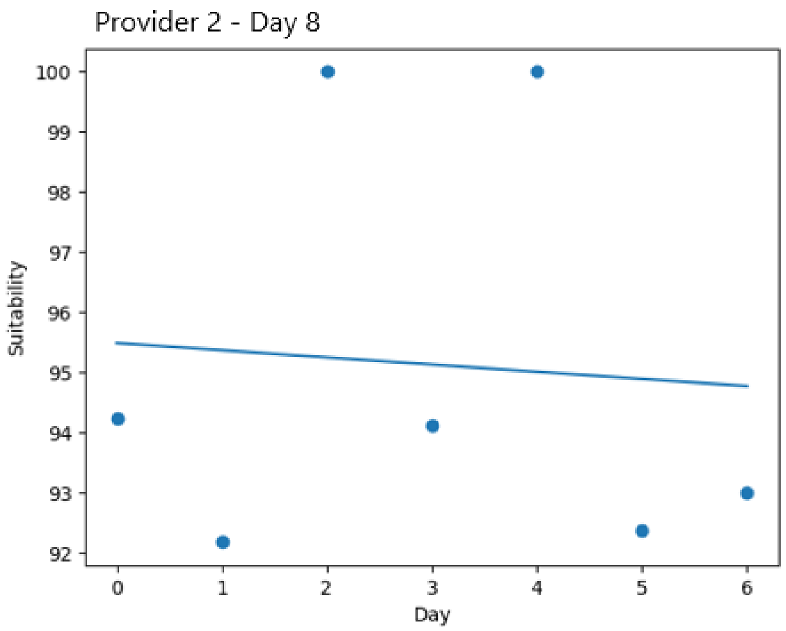
Definition: trendline of a scalar quantity’s (e.g. suitability) behavior over time

$$Y_i = \beta_0 + \beta_1 X_i$$

Use case: predicts the “future” suitability value

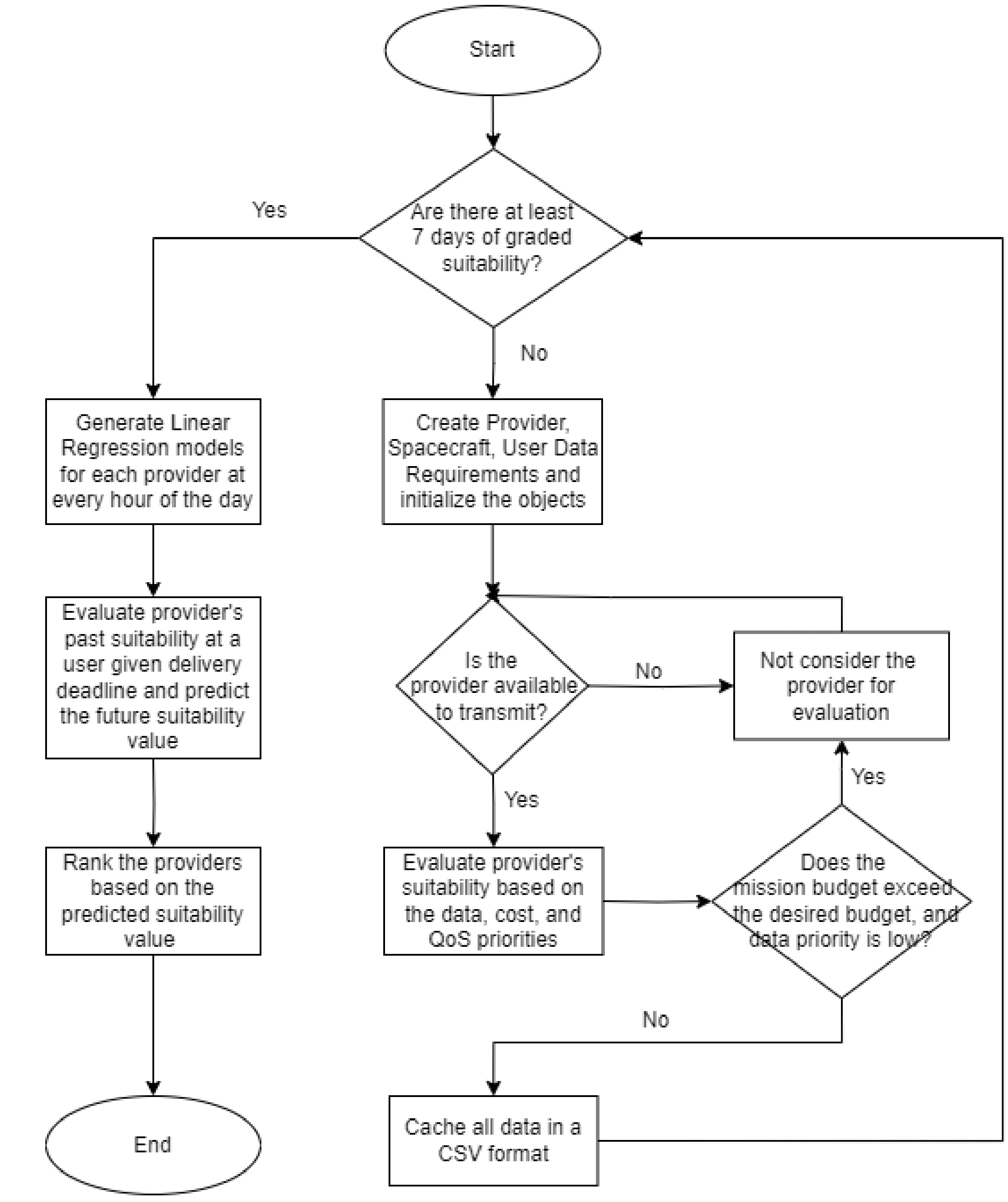
Advantage: suitability is now calculated based on a larger knowledge base of historic data

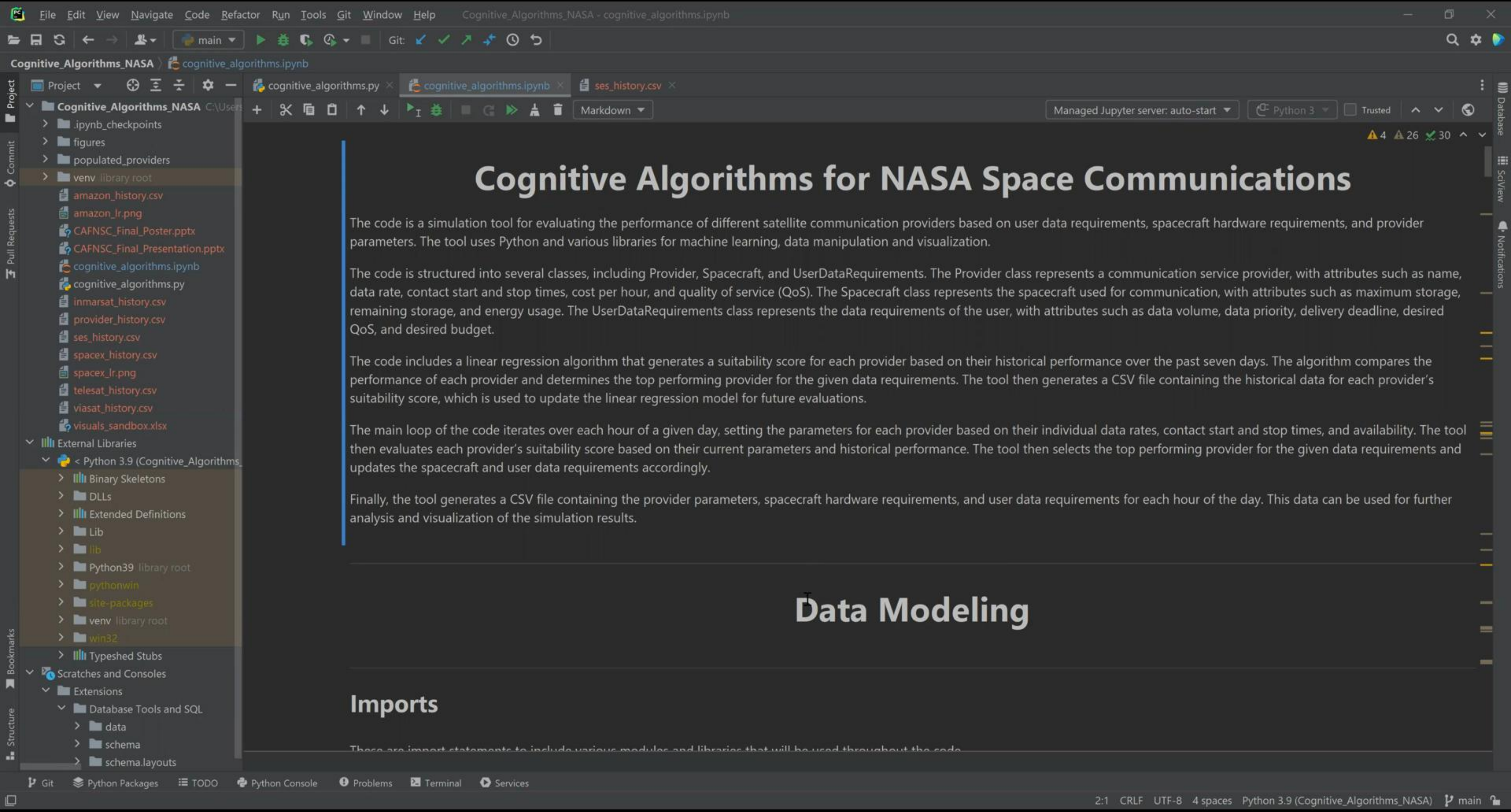




Top 1: Provider 2
Top 2: Provider 1

Flowchart of Data Collection and Algorithm Selection


Start
Are there at least 7 days of graded suitability?
Yes
No
Generate Linear Regression models for each provider at every hour of the day
Evaluate provider's past suitability at a user given delivery deadline and predict the future suitability value
Rank the providers based on the predicted suitability value
End
Create Provider, Spacecraft, User Data Requirements and initialize the objects
Is the provider available to transmit?
No
Not consider the provider for evaluation
Yes
Yes
Evaluate provider's suitability based on the data, cost, and QoS priorities
Does the mission budget exceed the desired budget, and data priority is low?
No
Cache all data in a CSV format



Cognitive Algorithms for NASA Space Communications

The code is a simulation tool for evaluating the performance of different satellite communication providers based on user data requirements, spacecraft hardware requirements, and provider parameters. The tool uses Python and various libraries for machine learning, data manipulation and visualization.

The code is structured into several classes, including Provider, Spacecraft, and UserDataRequirements. The Provider class represents a communication service provider, with attributes such as name, data rate, contact start and stop times, cost per hour, and quality of service (QoS). The Spacecraft class represents the spacecraft used for communication, with attributes such as maximum storage, remaining storage, and energy usage. The UserDataRequirements class represents the data requirements of the user, with attributes such as data volume, data priority, delivery deadline, desired QoS, and desired budget.

The code includes a linear regression algorithm that generates a suitability score for each provider based on their historical performance over the past seven days. The algorithm compares the performance of each provider and determines the top performing provider for the given data requirements. The tool then generates a CSV file containing the historical data for each provider's suitability score, which is used to update the linear regression model for future evaluations.

The main loop of the code iterates over each hour of a given day, setting the parameters for each provider based on their individual data rates, contact start and stop times, and availability. The tool then evaluates each provider's suitability score based on their current parameters and historical performance. The tool then selects the top performing provider for the given data requirements and updates the spacecraft and user data requirements accordingly.

Finally, the tool generates a CSV file containing the provider parameters, spacecraft hardware requirements, and user data requirements for each hour of the day. This data can be used for further analysis and visualization of the simulation results.

Data Modeling

Imports

These are import statements to include various modules and libraries that will be used throughout the code.

Future Works

- Balance evaluation parameters of suitability values to better reflect realistic grading

- Research a more optimal QoS metric for satellite service provider selection

- Balance the model to better handle outliers

Conclusions

- This implementation will output a ranking of available providers based upon calculated performances.

- Calculations account for user input as well as generated data describing behavior of providers

- Data generation uses our scope of familiarity to reflect realistic behavior of providers, do not represent real-world data

- We are modeling a recommender system, not recommending a certain provider

References

[1] J. Foust. NASA Selects Six Companies to Demonstrate Commercial Successors to TDRS. 2022. URL: https : / / spacenews . com / nasa - selects - six - companies - to – demonstrate-commercial-successors-to-tdrs/.

[2] S. Reddy. Cognitive Communication. 2020. URL: https: //sbir.nasa.gov/content/cognitive-communication-1.

[3] David Chelmins et al. "Cognitive communications for NASA space systems". In: Advances in Communications Satellite Systems. Proceedings of the 37th International Communications Satellite Systems Conference (ICSSC-2019). 2019, pp. 1–16. DOI: 10.1049/cp.2019.1222.

[4] Gilbert Clark et al. "Architecture for Cognitive Networking within NASA's Future Space Communications Infrastructure". In: Oct. 2016. DOI: 10 . 2514 / 6 . 2016-5725.

[5] Robin Burke, Alexander Felfernig, and Mehmet H. G ¨oker. "Recommender Systems: An Overview". In: AI Magazine 32.3 (June 2011), pp. 13–18. DOI: 10.1609/ aimag.v32i3.2361. URL: https://ojs.aaai.org/aimagazine/ index.php/aimagazine/article/view/2361.

[6] Robin Burke. "Hybrid Recommender Systems: Survey and Experiments". In: User Modeling and User-Adapted Interaction 12 (Nov. 2002). DOI: 10.1023/A: 1021240730564.

[7] B. Shetty. An In-Depth Guide to How Recommender Systems Work. 2019. URL: https : / / builtin . com / data - science/recommender-system

Q & A