Student Performance Prediction Pipeline

Priyanka Maurya

Objective & Problem Statement

Objective

Develop an early-warning predictive system that forecasts each student's final-test score from their demographics and daily habits, so that support can be deployed proactively—not reactively—before exams.

Problem Statement

Jumbled, inconsistent student records bury the few habits that predict exam success—so struggling learners aren't spotted until it's too late.

Why It Matters

Pinpointing at-risk students early—and focusing on the right behaviors—can boost test scores by up to 10 points.
Proactive, data-driven interventions can lift average scores by 5–10 points and reduce dropout rates.

Audience & Pain Points

Audience	**Pain Points**
School Principals & Academic Directors	• Fast, reliable risk metrics to allocate resources where they'll move the needle. • Flags appear only after grades drop, wasting precious remediation time
Student Welfare Officers & Counselors	• Limited tutoring and mentoring slots get wasted on low-impact cases. • Unclear which student habits to tackle first, so interventions feel scatter-shot. • By the time issues surface in reports, it's often too late to reverse the slide.

Dataset Overview

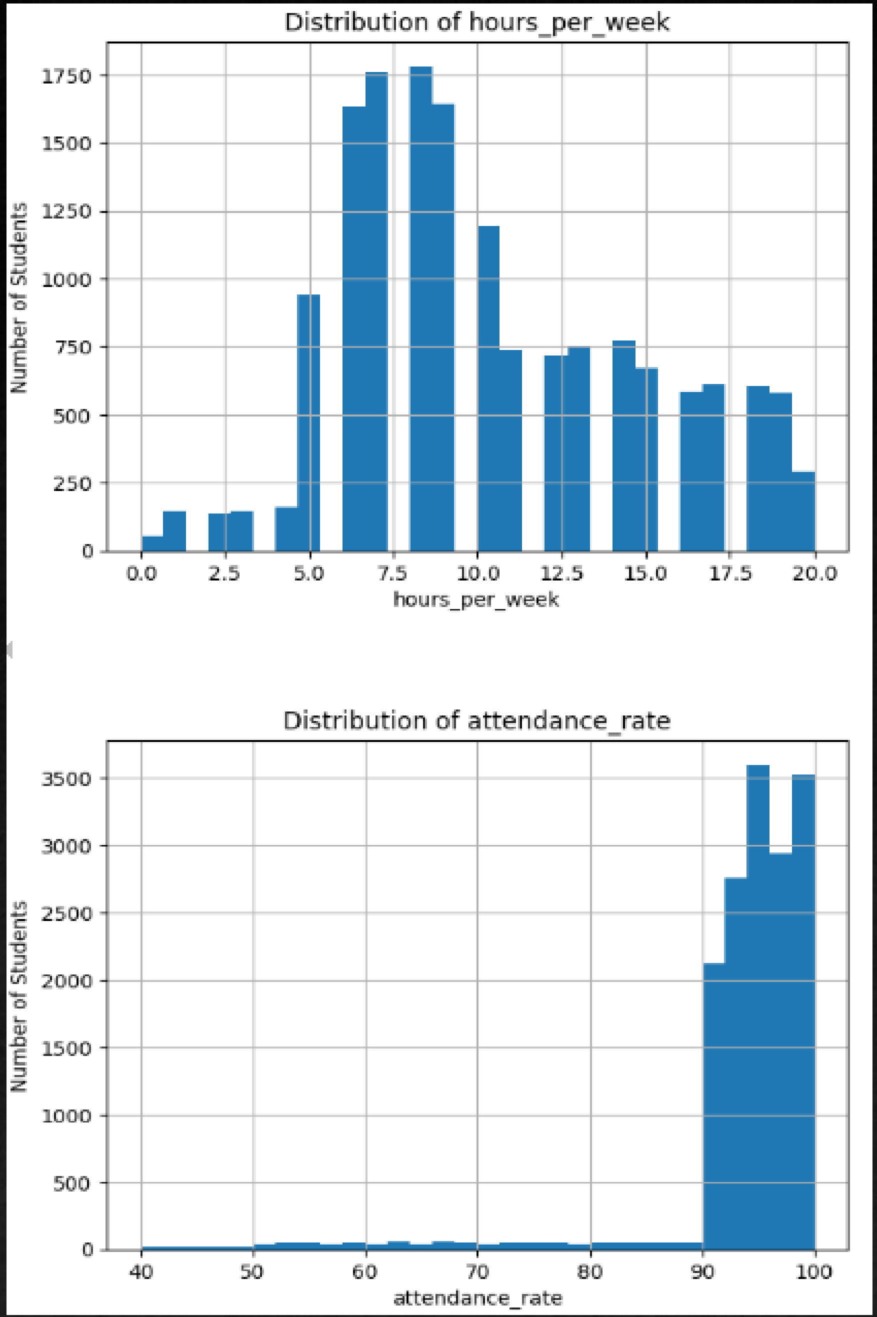
Leveraged over a **15,900** record student dataset—rich in both academic and lifestyle signals—to power my score-prediction model.

After cleansing the 1,273 gaps in data, I trained on 12,720 examples and held back 3,180 for testing to ensure robust performance.

18 core features	demographics, co-curricular activities, tuition status
Daily habits	attendance rate, hours studied, sleep & wake patterns
Missingness	only 1,273 total nulls across all fields
Train/Test split	80% (12,720) / 20% (3,180)
Target variable	final_test (math exam score)

index	number_of_siblings	direct_admission	CCA	learning_style	student_id	gender	tuition	final_test	n_male	n_female	age	hours_per_week	attendance_rate	sleep_time	wake_time	mode_of_transport	bag_color
0	0	Yes	Sports	Visual	ACN2BE	Female	No	69.0	14.0	2.0	16.0	10.0	91.0	22:00	6:00	private transport	yellow
1	2	No	Sports	Auditory	FGXIIZ	Female	No	47.0	4.0	19.0	16.0	7.0	94.0	22:30	6:30	private transport	green
2	0	Yes	None	Visual	B9AI9F	Male	No	85.0	14.0	2.0	15.0	8.0	92.0	22:30	6:30	private transport	white
3	1	No	Clubs	Auditory	FEVM1T	Female	Yes	64.0	2.0	20.0	15.0	18.0	NaN	21:00	5:00	public transport	yellow
4	0	No	Sports	Auditory	AXZN2E	Male	No	66.0	24.0	3.0	16.0	7.0	95.0	21:30	5:30	public transport	yellow
5	0	No	Arts	Visual	BA6R14	Female	No	57.0	9.0	12.0	15.0	11.0	96.0	22:30	6:30	private transport	red
6	2	Yes	None	Visual	D5WGTI	Male	No	69.0	12.0	3.0	16.0	15.0	93.0	21:30	5:30	public transport	green
7	0	No	Sports	Visual	HTP8CW	Male	No	76.0	20.0	2.0	15.0	3.0	97.0	21:00	5:00	public transport	green
8	0	No	Arts	Auditory	U3YRTC	Male	No	57.0	20.0	7.0	15.0	15.0	98.0	22:00	6:00	private transport	red
9	2	No	Arts	Auditory	3MOMA6	Male	Yes	60.0	13.0	9.0	16.0	16.0	NaN	22:30	6:30	private transport	green

Discoveries from EDA
(Exploratory Data Analysis)

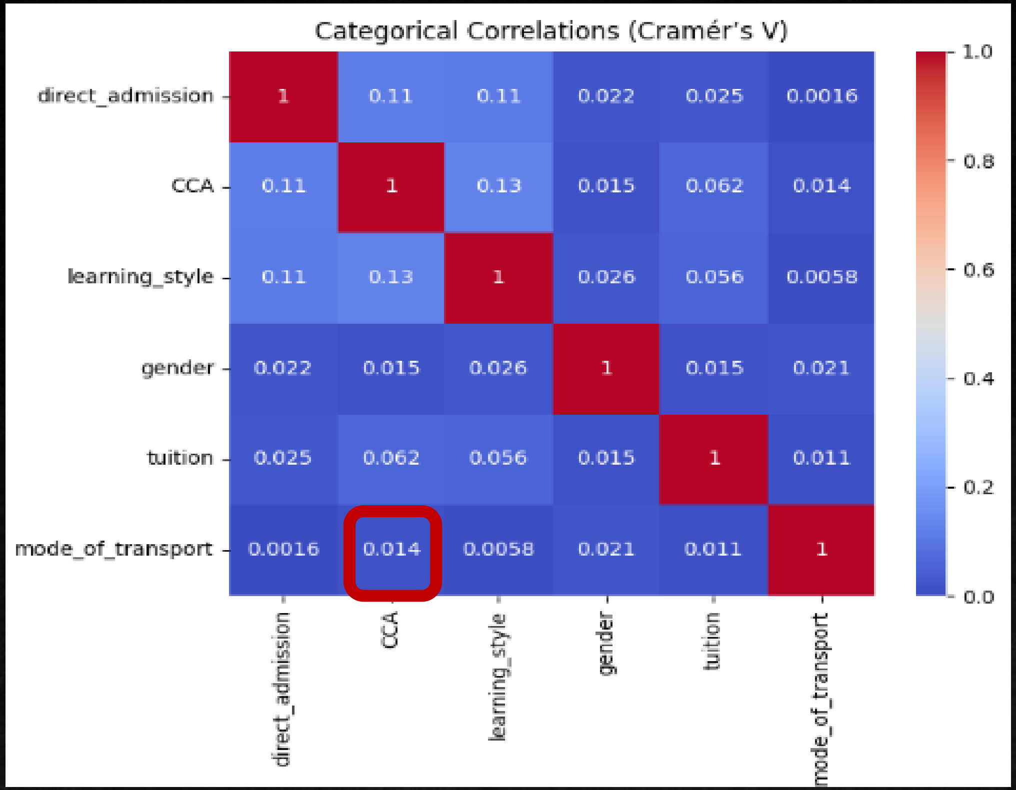


"Most students study 5–10 hours/week"

Study hours range from 0 to 20 per week, clustering around 6–8 hours—indicating a healthy spread

"Clustered tightly at 90–100%"

A small minority of students record significantly lower attendance rates.



Categorical features act largely independently (V<0.14)

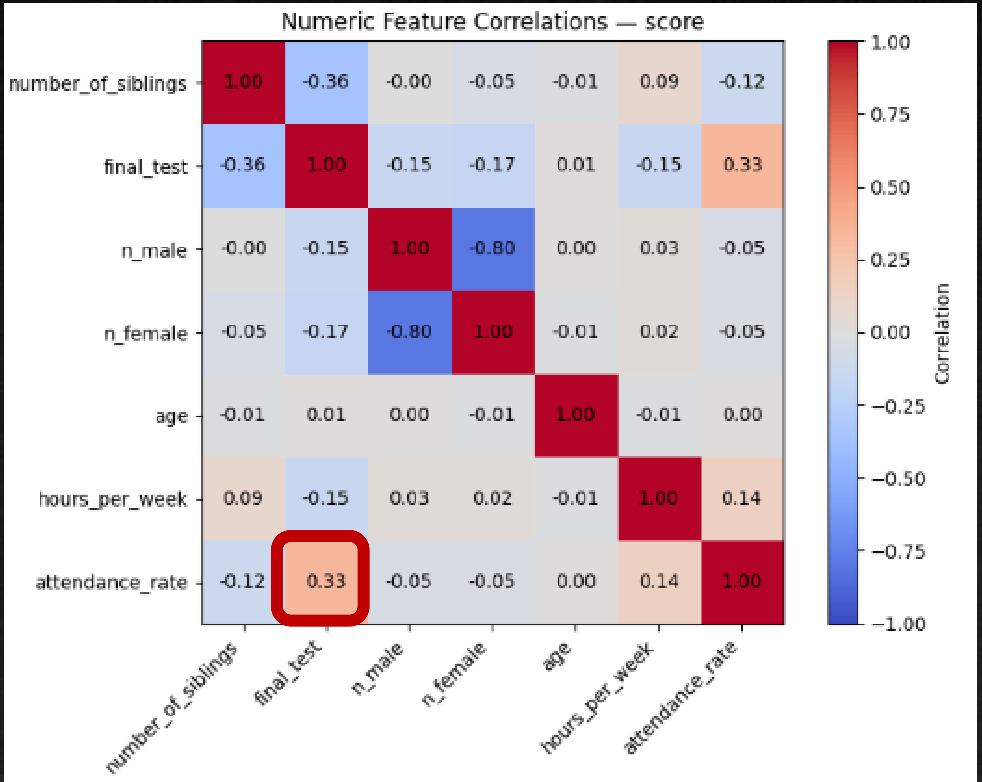
Whether a student is in tuition, a particular CCA, or a given learning style has almost no overlap—each behaves as its own signal

Cramér's V plots quantify and visualize feature relationships at a glance, guiding effective feature selection for modeling.

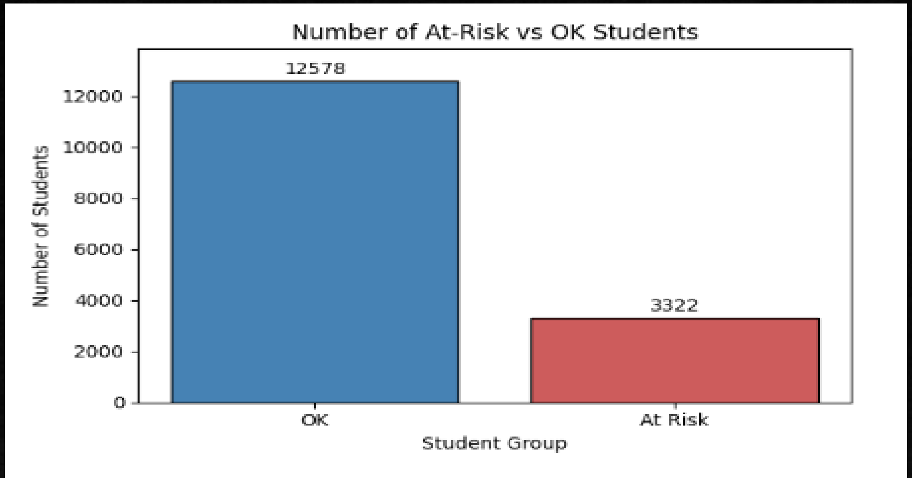
Quantifies categorical relationship strength: 0 means unrelated, 1 means perfectly aligned.

Attendance shows the strongest link (r≈0.33)

Students who boost attendance from 80–90% into the 90–100% bin see, on average, a 5–10-point lift in their final-test scores



Key Take Away



At-Risk vs OK

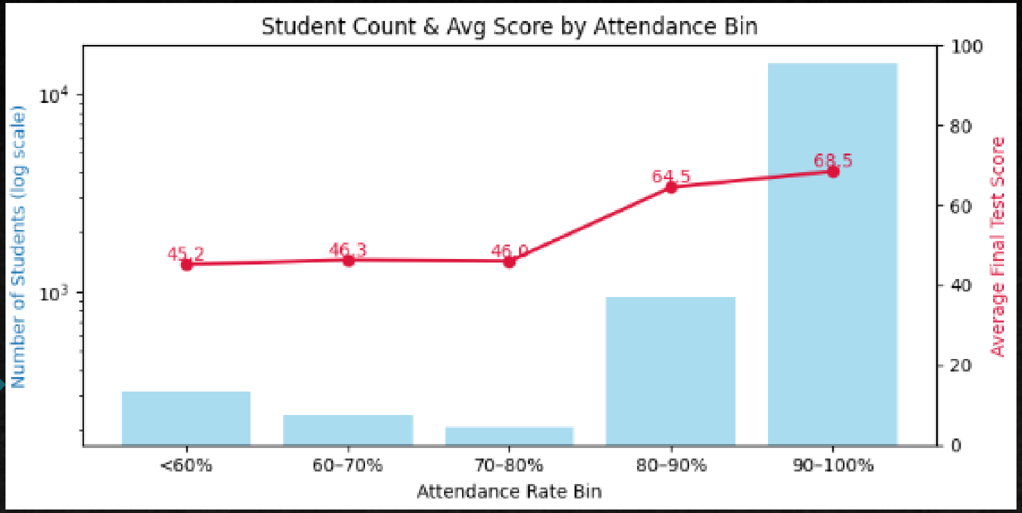
3322 students (21 %) = **At Risk** & 12578 (79 %) = **OK**

Students scoring at or below the 20th percentile on the final test (the bottom quintile) are flagged as "At Risk."

"Scores surge once attendance tops 80 %"

Below 80 % attendance, students cluster around 35–50 points; 80–90 % median jumps to ~65, and at 90–100 % it reaches ~68, with top scores at 100.

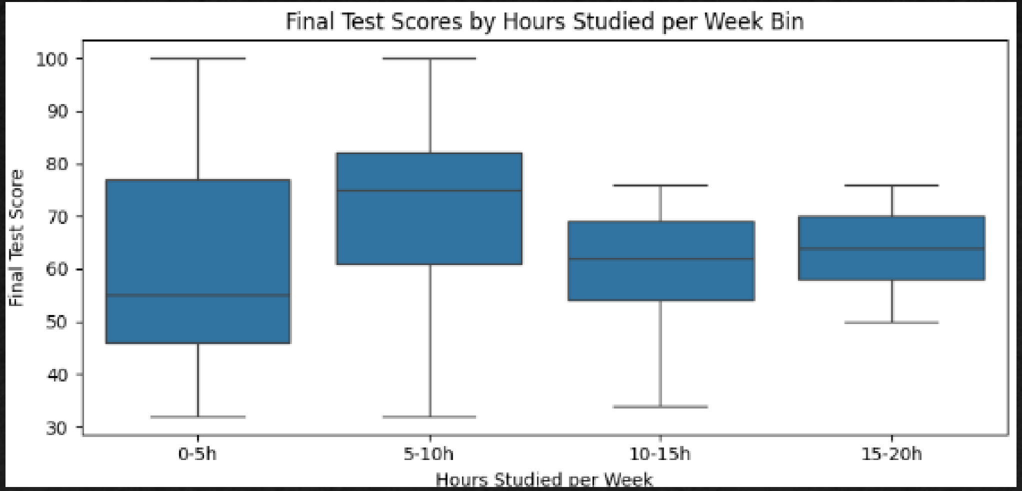
Better attendance drives stronger performance.



"5–10 h/week study hits the sweet spot"

Those studying 5–10 hours weekly boast the highest median (~75), whereas 0–5 h lags (~50) and 10–20 h yields only modest gains (~60–65)

Study quantity alone doesn't guarantee higher scores.



Machine Learning Pipeline

Show me the data
(eda.ipynb)

—

interactive charts and stats uncover which student habits matter most

Grab the records
(src/load_data.py)

—

pulls all student info from the SQLite database.

Clean and prep
(src/preprocess.py)

—

fixes typos, turns categories into numbers, & scales everything so the model can learn.

Teach the machine
(src/train.py)

—

fits a Random Forest to learn how habits predict test scores.

Check the results
(src/evaluate.py)

—

measures accuracy (RMSE, R^2) and saves the trained model for future use

Key Results & Interpretation

🏅 **RMSE 7.48**

on average, model's predictions are within ± 7½ points of actual scores

📊 **R^2 0.70**

it explains 70 % of the variability in student performance

"In other words, model predicts each student's final score with an average error of ±7.5 points—capturing 70 % of the score variation—and flags at-risk learners well before exam day"

Goal Hit & Pain Solved

- **Unified & cleaned** 15 900 records across 18 features—typo fixes, label standardization (e.g. "Y"→"Yes," "CLUBS"→"Clubs"), and sensible imputations.
- **Rigorous EDA distilled key drivers**:
  - Attendance : moving from 80–90 % to 90–100 % boosts scores by 5–10 points.
  - Sleep duration: 7–8 h correlates with top results.
- **Built an 80/20 train/test pipeline using a 50-tree Random Forest**:
  - Flags the bottom 20 % (~ 3 322 students) as "At Risk" for early support.
  - Splits data (12 720 train / 3 180 test) with one script.
- **Reproducible, one-command workflow** (load → preprocess → train → evaluate → predict) ensures every semester can run the model with zero manual steps.
- **Actionable output**: Automated alerts that cut "time-to-flag" by weeks—focusing interventions on the right students and behaviors and yielding 5–10 point score improvements.

Q&A

Collaboration

- Invited collaborators to GitHub to view/run the code
- Used run.sh to execute end-to-end workflow
- Included a detailed README.md outlining project setup, repository structure, and end-to-end execution steps.

Next Steps

- Test additional models (e.g. XGBoost)
- Integrate into CI/CD for automated retraining.

Thank You

Contact: https://github.com/rush2priyanka

Project Repository: https://github.com/rush2priyanka/Challenge_1.git