

HW4 - generate prediction challenge

Nam Lai - Data101- 12/5/2025

Prediction Challenge: Modeling Social Media Burnout Among Students

In this project, I design a **prediction challenge** based on a student survey dataset about social media use, sleep, and mental health.

Instead of predicting a simple numeric outcome (like hours of usage), I create a new **3-class target**:

$$\text{Social_Media_Burnout} \in \{\text{Low, Moderate, High}\}$$

Intuitively, Social Media Burnout represents how much a student's **daily life is negatively affected** by social media overuse: feeling exhausted, drained, or overwhelmed.

However, this target is **not originally present in the dataset**. I construct it by:

1. Resampling and scrambling a real survey to create a larger, more anonymous dataset.
2. Creating a **hidden continuous "risk score"** that depends on a small set of meaningful factors:
 - sleep debt
 - heavy social media usage
 - mental health struggles
 - conflicts caused by social media
 - plus some effects from academic level, relationship status, and platform
3. Converting that risk score into 3 groups: Low / Moderate / High burnout.

This allows me to design a challenge where:

- Simple off-the-shelf models (like a basic decision tree or multinomial logistic regression on the raw variables) **do fine but not great**
- A more thoughtful approach that **reconstructs the hidden structure** (by engineering features like Sleep_Debt, Usage_Over_4, etc.) plus a stronger model (random forest) can reach **much higher accuracy**

This matches the goal of the assignment: the patterns should be **non-trivial but discoverable** with good EDA, feature engineering, and the right model choice

Dataset Schema:

- **Source and size:**
 - Original source: Students_Social_Media_Addiction.csv (a real survey about students' social media use, sleep, conflicts, and mental health).
 - I expanded it to a **synthetic dataset with 10,000 rows** using bootstrap resampling.
 - Then I split into:
 - **Training set:** students_burnout_train.csv ($\approx 9,000$ rows, 90%)
 - **Testing set:** students_burnout_test.csv ($\approx 1,000$ rows, 10%)
 - Each row corresponds to one synthetic student

- Variables exposed to students: The final CSV files contain the following columns

Variable Name	Type	Range / Categories (from data)	Description
Student_ID	Integer		Synthetic row identifier created after resampling. Used only for bookkeeping and dropped before modeling.
Age	Integer (numeric)	18–24	Student’s age in years. (In this synthetic sample, all observations are traditional college-age students.)
Gender	Categorical	{Female, Male}	Self-reported gender.
Academic_Level	Categorical	{High School, Undergraduate, Graduate}	Student’s educational level; also used in the burnout rule system.
Country	Categorical	110 distinct countries (e.g., <i>China, India, Mexico, Poland, South Korea, ...</i>)	Country of residence. This variable is not used in the label-generation rules and serves as pure noise in the prediction task.
Avg_Daily_Usage_Hours	Numeric (float)	1.5 – 8.5 hours/day	Average daily time spent on social media. Values come from the original survey and are scrambled across rows.

Sleep_Hours_Per_Night	Numeric (float)	3.8 – 9.6 hours/night	Typical number of hours the student sleeps per night. Used to compute hidden “sleep debt” features.
Mental_Health_Score	Integer	4 – 9	Self-reported mental health rating; higher values indicate better mental well-being.
Conflicts_Over_Social_Media	Integer	0 – 5	Frequency or intensity of conflicts that arise because of social media use.
Addicted_Score	Integer	2 – 9	Self-perceived addiction to social media.
Most_Used_Platform	Categorical	{Facebook, Instagram, KakaoTalk, LINE, LinkedIn, Snapchat, TikTok, Twitter, VKontakte, WeChat, WhatsApp, YouTube}	The platform the student reports using most often. Certain platforms (e.g., Instagram/TikTok, LinkedIn) are given special roles in the hidden burnout rules.
Relationship_Status	Categorical	{Single, In Relationship, Complicated}	Current relationship status. For undergraduates, this interacts with conflict levels in the burnout rule system.

Social_Media_Burnout	Categorical (target)	{Low, Moderate, High}	Target variable to be predicted. This label is synthetically generated from hidden risk rules based on sleep, usage, mental health, conflicts, academic level, relationship status, and platform, plus a small amount of random noise.
----------------------	-------------------------	--------------------------	--

- **Hidden / internal variables (not in the released CSV):** Inside the generation code, I define several engineered features that are not visible to students but drive the label:
 - Usage_to_Sleep_Ratio = Avg_Daily_Usage_Hours / Sleep_Hours_Per_Night
 - Sleep_Debt = max(0, 7 – Sleep_Hours_Per_Night)
 - How much the student sleeps less than 7 hours.
 - Usage_Over_4 = max(0, Avg_Daily_Usage_Hours – 4)
 - How many hours per day are above a “safe” baseline of 4 hours.
 - Ratio_Excess = max(0, Usage_to_Sleep_Ratio – 0.8)
 - Whether usage is high relative to available sleep.
 - Mental_Struggle = max(0, 6 – Mental_Health_Score)
 - How much the mental health score falls below a threshold of 6.
 - Conflict_Intensity = Conflicts_Over_Social_Media
 - Direct count or intensity of conflicts caused by social media.

These hidden features are used to compute a continuous risk_score, which then gets turned into the burnout label

How the dataset is generated:

- Expanding the original dataset with bootstrap sampling:

I generate 10,000 synthetic rows by sampling with replacement from the original dataset:

```
set.seed(2025)
```

```
setwd("/Users/lainam/Desktop")
```

```
df_raw <- read.csv("Students_Social_Media_Addiction.csv",
  stringsAsFactors = FALSE)
```

```
N_ROWS <- 10000
```

```
idx <- sample(seq_len(nrow(df_raw)), size = N_ROWS, replace = TRUE)
```

```
df <- df_raw[idx, ]  
rownames(df) <- NULL
```

```
df$Student_ID <- seq_len(nrow(df))
```

-> Maintains realistic **marginal distributions** (for usage, sleep,...) but gives a larger dataset for building and testing models

- If I stopped at bootstrap, the original correlations from the survey would remain and could dominate any new rules I add

-> So I **scramble** several key columns: I randomly permute them across rows (without replacement), which keeps the same values overall but destroys their original relationships

```
cols_to_scramble <- c(  
  "Country",  
  "Avg_Daily_Usage_Hours",  
  "Most_Used_Platform",  
  "Affects_Academic_Performance",  
  "Sleep_Hours_Per_Night",  
  "Mental_Health_Score",  
  "Relationship_Status",  
  "Conflicts_Over_Social_Media",  
  "Addicted_Score"  
)
```

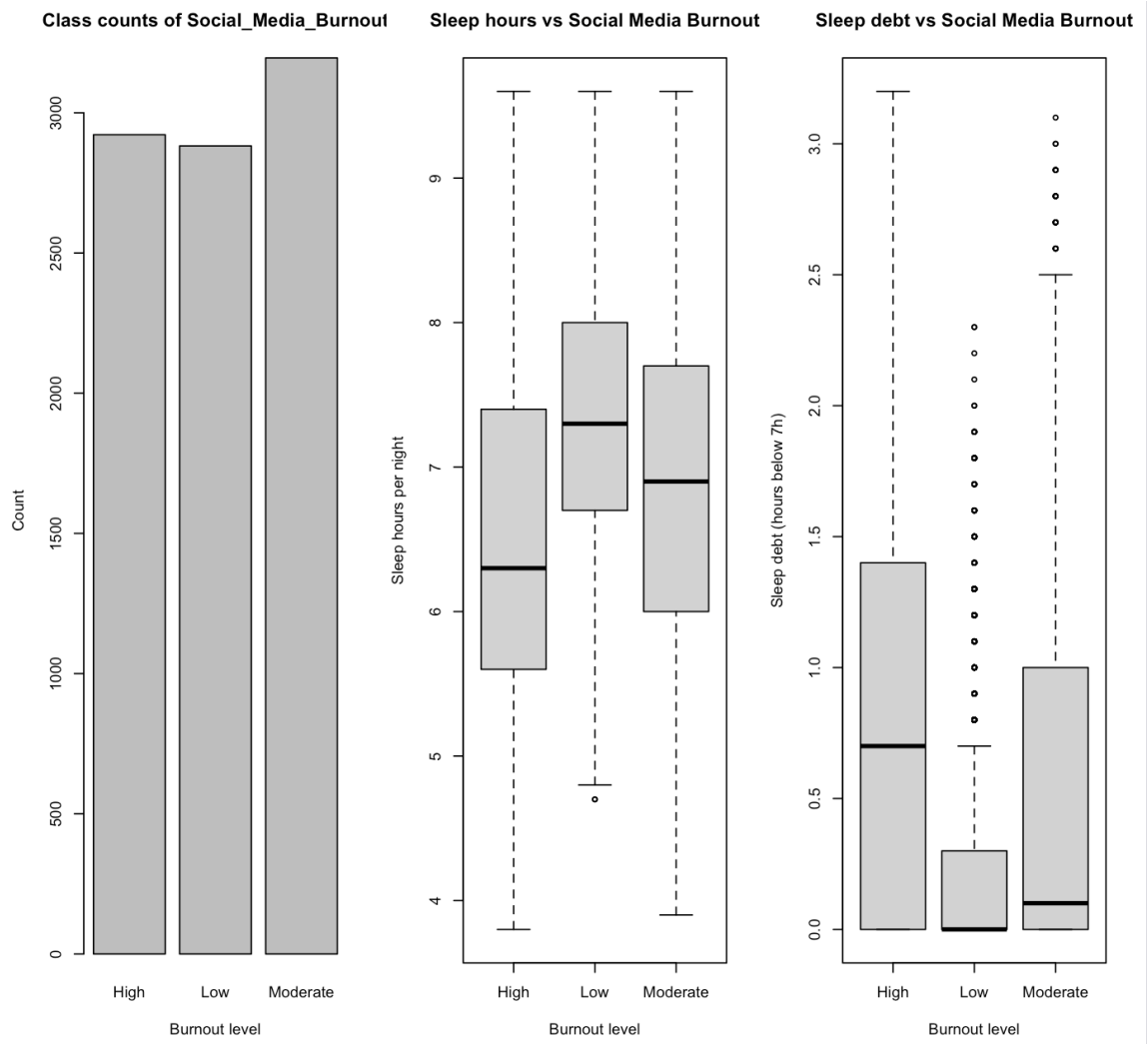
```
for (nm in cols_to_scramble) {  
  if (nm %in% names(df)) {  
    df[[nm]] <- sample(df[[nm]], size = nrow(df), replace = FALSE)  
  }  
}
```

-> The **distribution** of each column stays realistic

-> Original joint patterns are **destroyed**

-> This gives me a clean structure to paint my own synthetic burnout rules on top

- Sleep_Debt: How much rest the student is missing



`Sleep_Debt <- pmax(0, 7 - df$Sleep_Hours_Per_Night)`

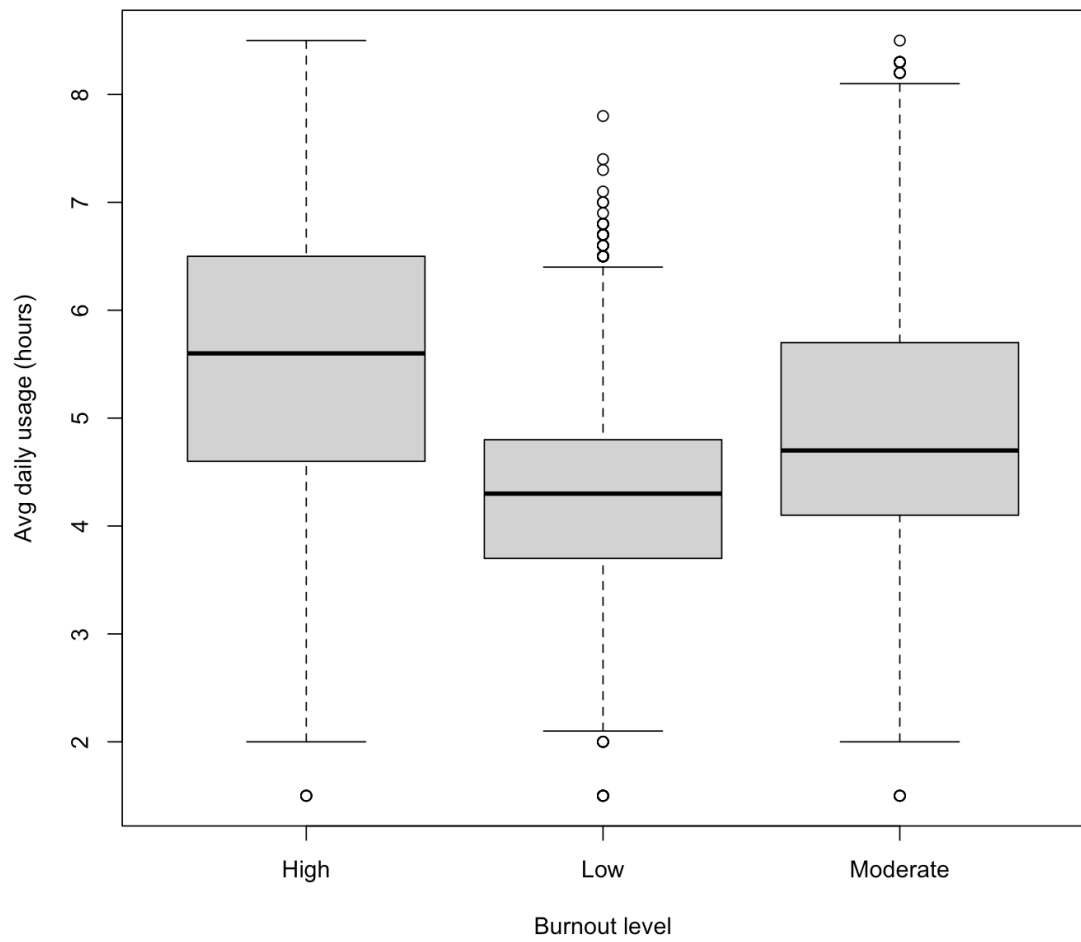
This feature measures how many hours below the recommended 7 hours of sleep a student gets. Research consistently shows that insufficient sleep intensifies emotional exhaustion, worsens concentration, and increases burnout symptoms.

- Someone sleeping **5 hours/night** has a Sleep_Debt of **2**
- Someone sleeping **8 hours/night** has a Sleep_Debt of **0**

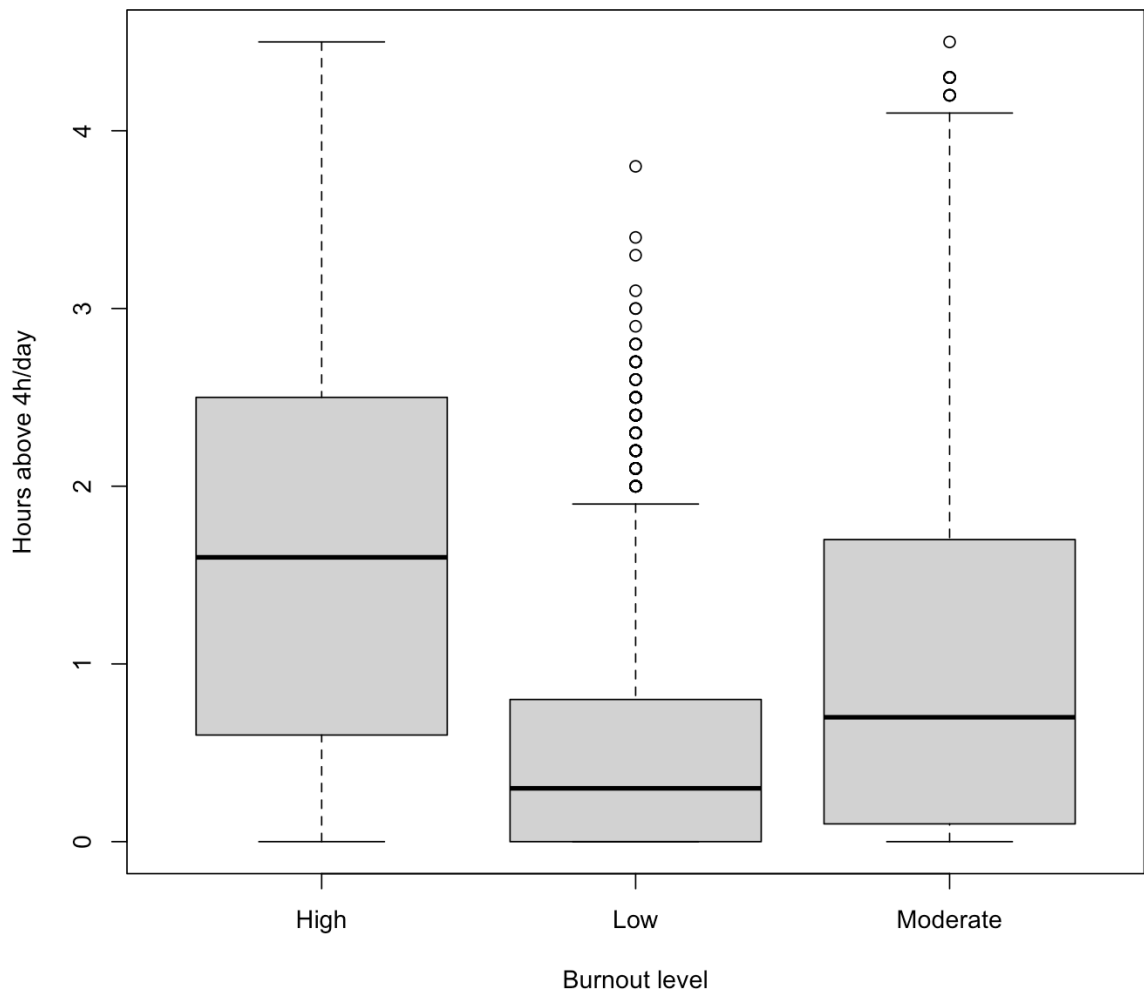
This feature directly drives burnout in all three academic levels.

- Usage_Over_4: Excessive daily social media usage

Daily social media usage vs Burnout



Usage above 4 hours vs Burnout



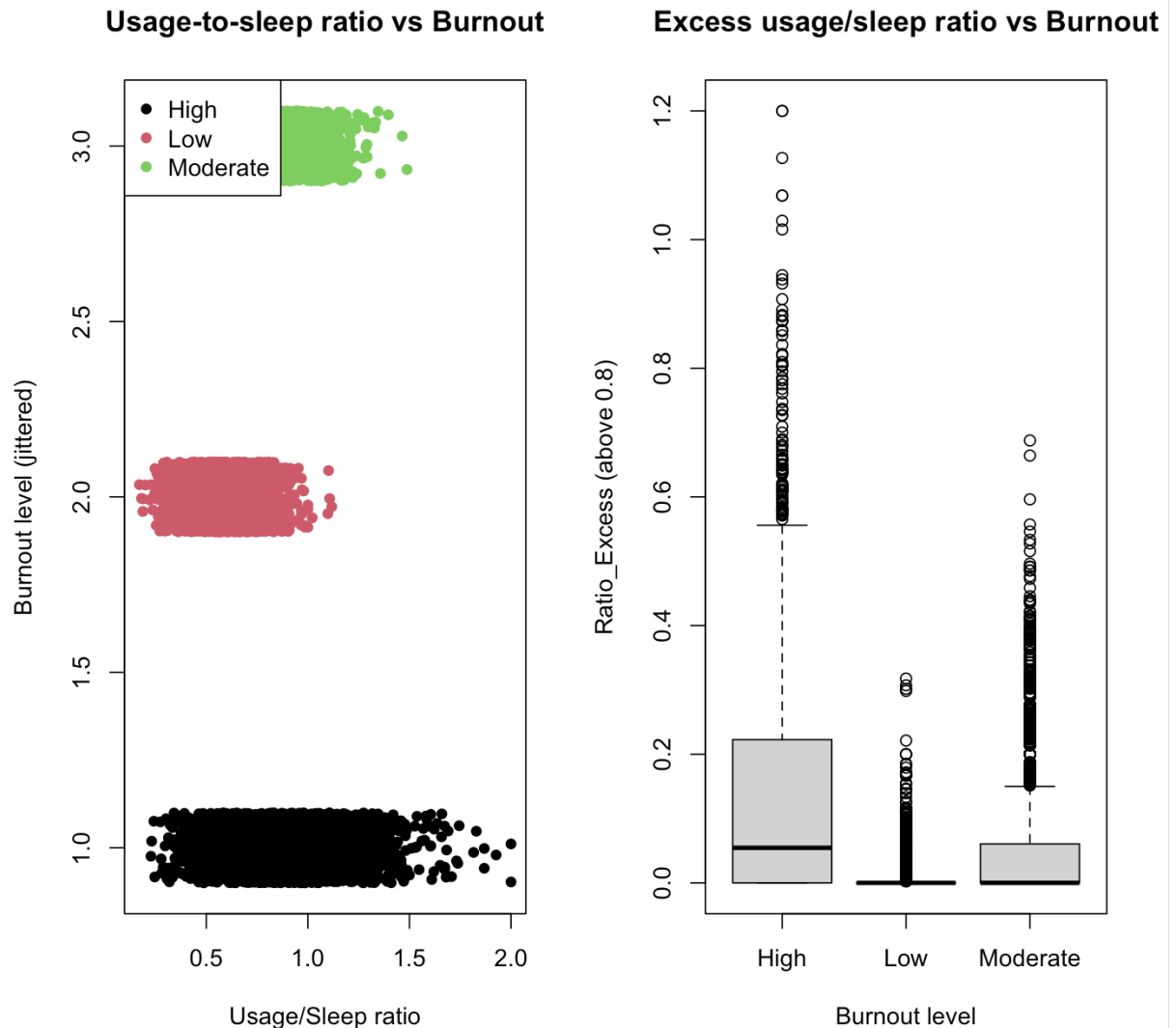
```
Usage_Over_4 <- pmax(0, df$Avg_Daily_Usage_Hours - 4)
```

The idea is that *some* social media use is normal, but after around **4 hours per day**, psychological studies show diminishing well-being and increased stress

- A student who uses social media 6.5 hours/day has **2.5 hours** of “excessive usage.”
- If usage ≤ 4 hours, the value is **0**, meaning “healthy usage.”

This feature is strongly weighted for High School and Undergraduate students.

- Ratio_Excess: When social media use grows faster than sleep



```
Usage_to_Sleep_Ratio <- df$Avg_Daily_Usage_Hours / df$Sleep_Hours_Per_Night
```

```
Ratio_Excess <- pmax(0, Usage_to_Sleep_Ratio - 0.8)
```

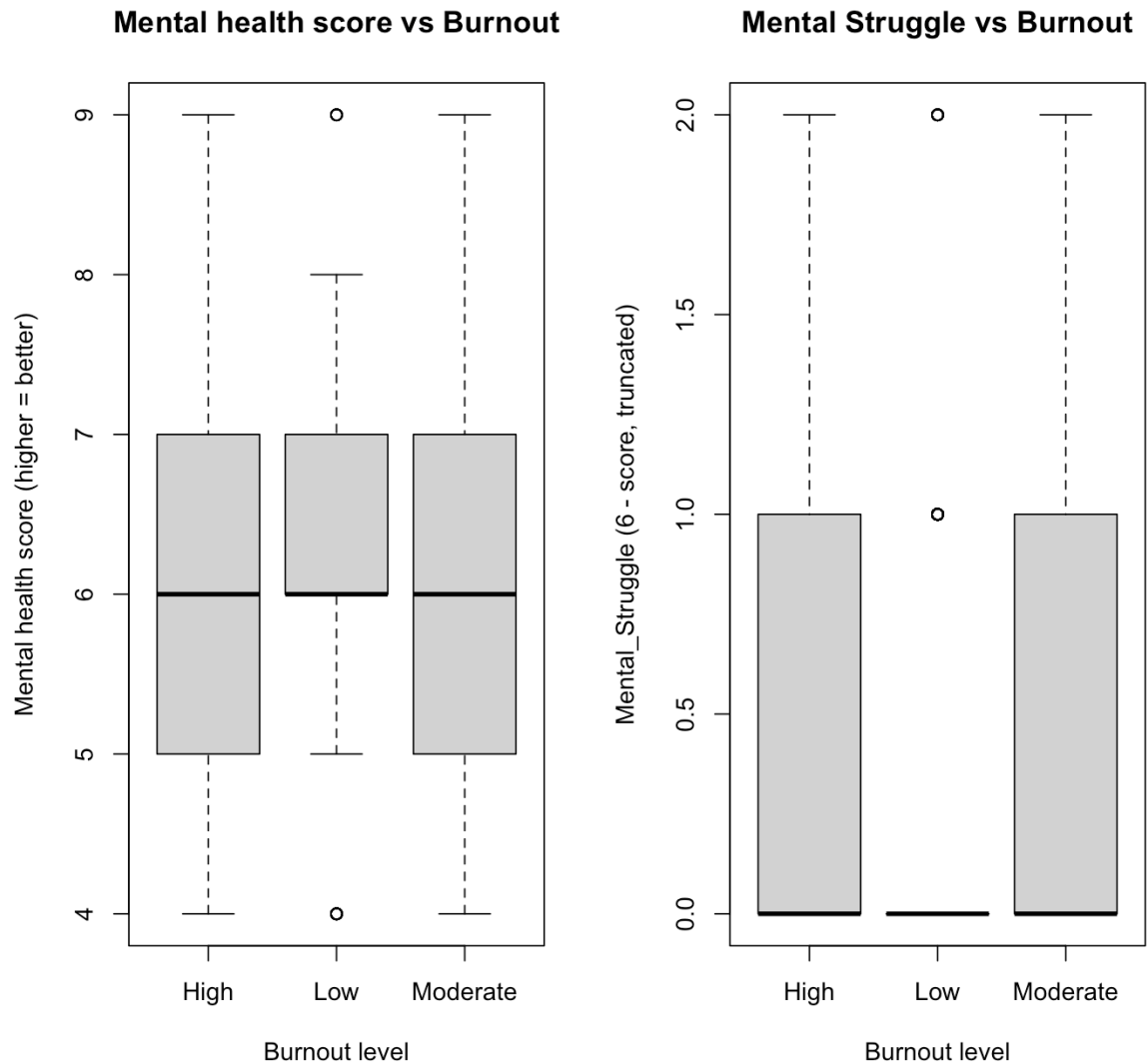
This feature captures the idea that **burnout is not only caused by heavy usage or low sleep individually, but by the imbalance between them.**

Example:

- Student A: 4 hours usage / 8 hours sleep = 0.50 → healthy
- Student B: 6 hours usage / 5 hours sleep = 1.20 → ratio imbalance → high burnout risk

Only usage above the 0.8 ratio threshold contributes to burnout, creating a *nonlinear* “point” that simple models may miss

- Mental_Struggle: The inverse of mental health



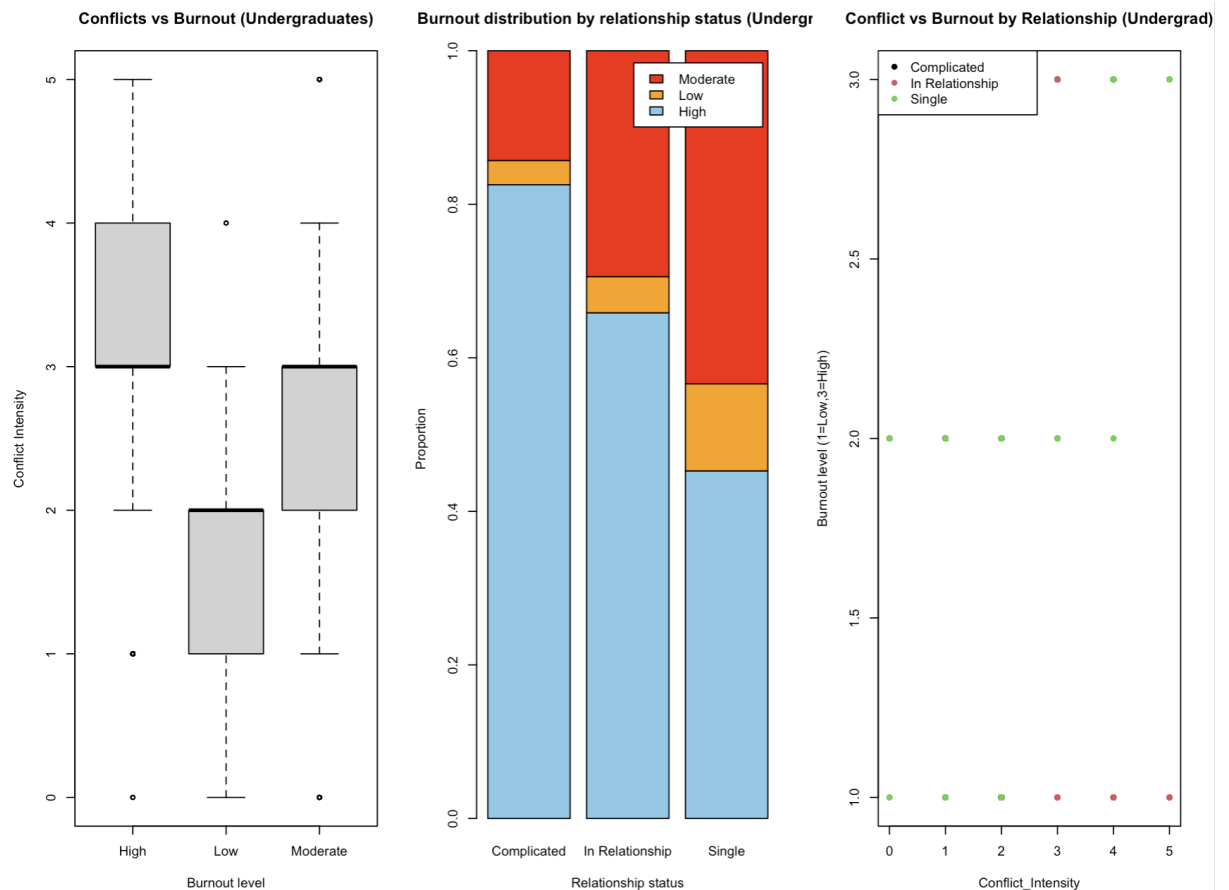
```
Mental_Struggle <- pmax(0, 6 - df$Mental_Health_Score)
```

Higher mental health scores mean better well-being, so this feature reverses the scale:

- Score 9 → Mental_Struggle = 0 (healthy)
- Score 4 → Mental_Struggle = 2 (high psychological strain)

This feature is especially important for **Graduate students**, whose burnout rules rely heavily on mental well-being

- Conflict_Intensity — Social friction as an early warning signal



```
Conflict_Intensity <- df$Conflicts_Over_Social_Media
```

-> This feature is intentionally used as-is (not thresholded) because conflicts tend to escalate burnout in a roughly linear way

-> Undergraduates are particularly sensitive to interpersonal issues, so this feature is weighted more heavily for them

-> This variable also participates in relationship-status interactions

- Relationship Interaction Terms (Undergraduates Only):

```
risk_score += 0.3 * Conflict_Intensity (for In Relationship only)
```

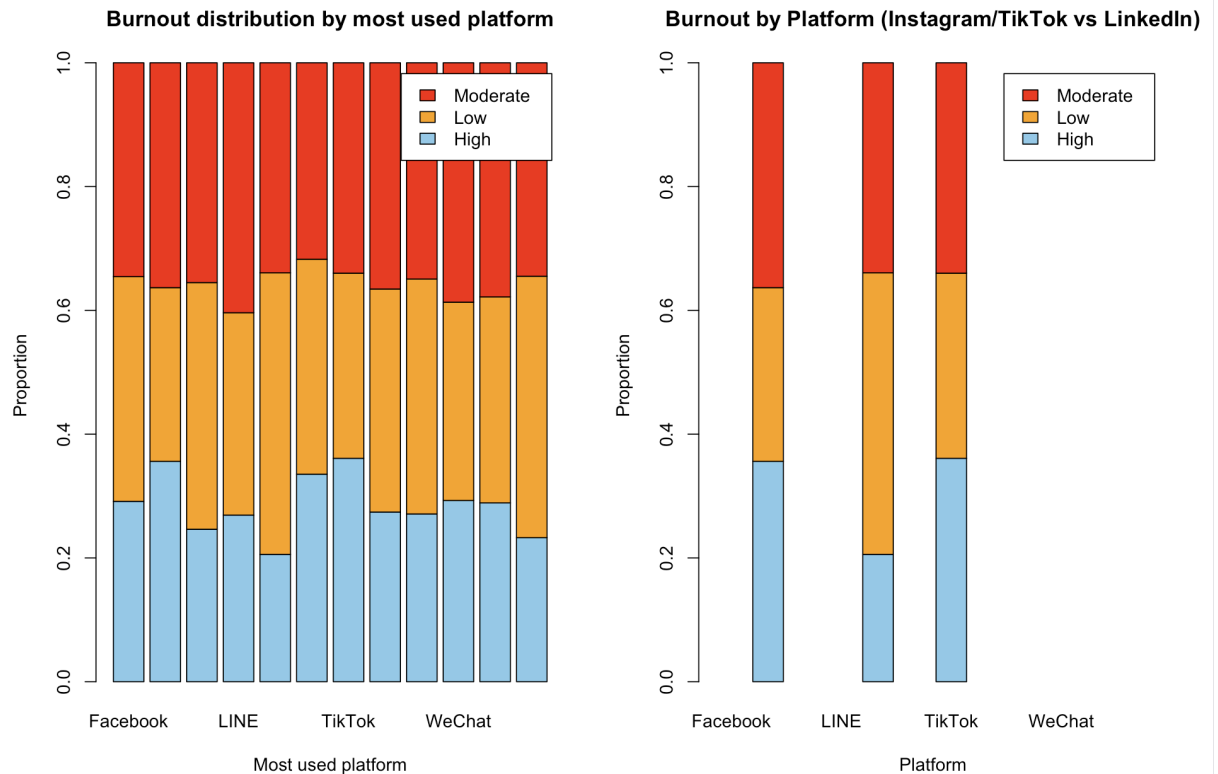
```
risk_score += 0.6 * Conflict_Intensity + 0.3 * Mental_Struggle
```

These rules embed realistic behavior:

- Students in stable relationships experience burnout when conflicts arise.
- Students in “Complicated” situations experience even stronger emotional swings.

These interactions are intentionally included because they create **non-linearity** and **population-specific structure** that simple linear models fail to capture.

- Platform effects: Instagram/TikTok: more High burnout, LinkedIn: more Low/Moderate, less High

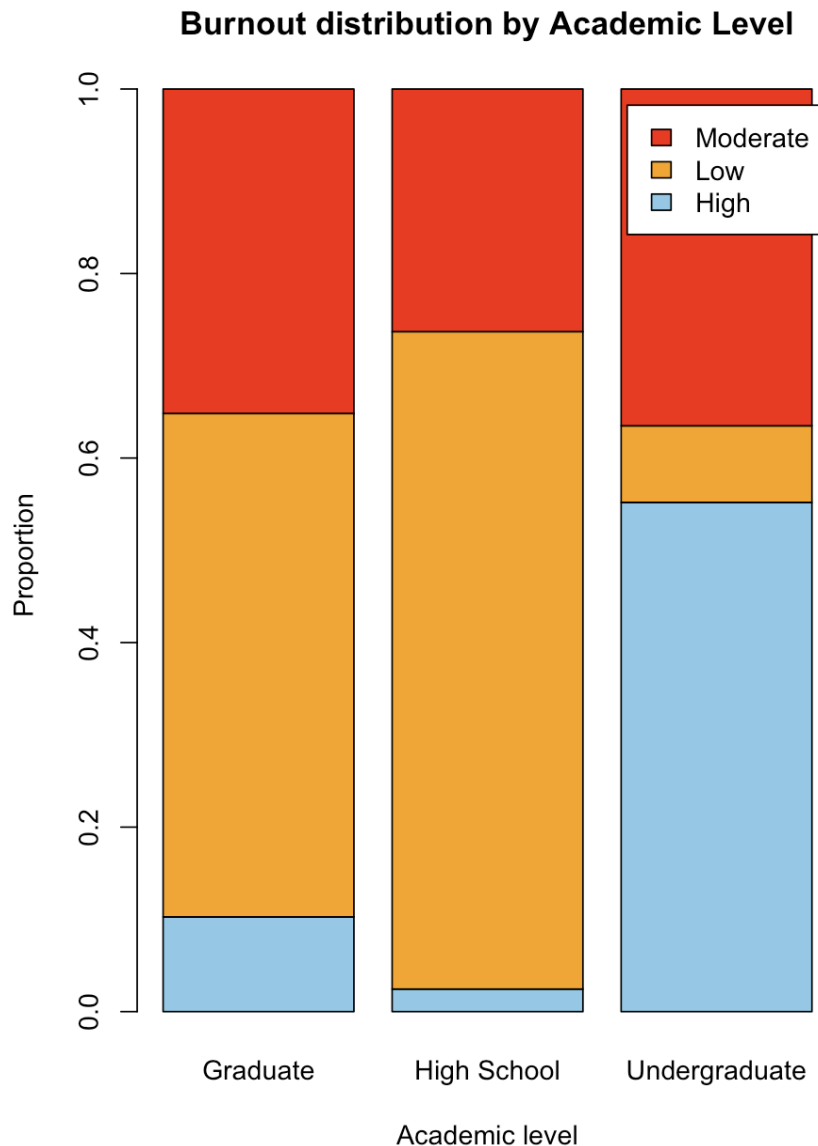


```
tab_plat <- table(train$Most_Used_Platform, train$Social_Media_Burnout)
prop_plat <- prop.table(tab_plat, margin = 1)
```

```
barplot(t(prop_plat),
  beside = FALSE,
  col = c("skyblue", "orange", "red"),
  legend.text = TRUE,
  main = "Burnout distribution by most used platform",
  xlab = "Most used platform",
  ylab = "Proportion")
```

- Academic-level-specific rules for risk_score:


```
is_hs <- df$Academic_Level == "High School"
is_ug <- df$Academic_Level == "Undergraduate"
is_gr <- df$Academic_Level == "Graduate"
```



- High School students: High school burnout is mostly about **sleep loss and heavy usage**, with some extra penalty if usage/sleep ratio is high.

```
risk_score[is_hs] <-
  0.9 * Sleep_Debt[is_hs] +
  0.7 * Usage_Over_4[is_hs] +
  0.4 * Ratio_Excess[is_hs]
```

- Undergraduates: burnout is a **mix** of sleep debt, heavy usage, usage/sleep ratio, mental health struggles, and conflicts

```
risk_score[is_ug] <-
  1.0 * Sleep_Debt[is_ug] +
  0.8 * Usage_Over_4[is_ug] +
  0.7 * Ratio_Excess[is_ug] +
  0.9 * Mental_Struggle[is_ug] +
  0.7 * Conflict_Intensity[is_ug]
```

```
in_rel_ug <- is_ug & df$Relationship_Status == "In Relationship"
risk_score[in_rel_ug] <- risk_score[in_rel_ug] +
  0.3 * Conflict_Intensity[in_rel_ug]
```

```
comp_ug <- is_ug & df$Relationship_Status == "Complicated"
risk_score[comp_ug] <- risk_score[comp_ug] +
  0.6 * Conflict_Intensity[comp_ug] +
  0.3 * Mental_Struggle[comp_ug]
```

-> Being "In Relationship" or especially "Complicated" **magnifies** the effect of conflicts and poor mental health

- Graduate students: For grad students, **mental health and sleep** are even more important, Social media usage still matters, but slightly less than mental factors

```
risk_score[is_gr] <-
  1.1 * Sleep_Debt[is_gr] +
  1.1 * Mental_Struggle[is_gr] +
  0.8 * Usage_Over_4[is_gr] +
  0.4 * Ratio_Excess[is_gr]
```

- From risk_score to 3-level burnout with noise:

```
q_low <- as.numeric(quantile(risk_score, probs = 0.33))
q_high <- as.numeric(quantile(risk_score, probs = 0.67))
```

```
burnout_det <- ifelse(
  risk_score < q_low, "Low",
  ifelse(risk_score < q_high, "Moderate", "High")
)
```

```
burnout_det <- factor(burnout_det,
  levels = c("Low", "Moderate", "High"))
```

```

set.seed(2025)

flip_idx <- sample(seq_len(n), size = round(0.05 * n))

burnout_noisy <- burnout_det

for (i in flip_idx) {
  current <- burnout_det[i]

  if (current == "Low") {
    burnout_noisy[i] <- "Moderate"
  } else if (current == "High") {
    burnout_noisy[i] <- "Moderate"
  } else if (current == "Moderate") {
    if (runif(1) < 0.5) {
      burnout_noisy[i] <- "Low"
    } else {
      burnout_noisy[i] <- "High"
    }
  }
}

df$Social_Media_Burnout <- factor(burnout_noisy,
                                  levels = c("Low", "Moderate", "High"))

```

This gives about one-third of students in each class. Making the dataset more realistic and avoid 100% deterministic labels, I add 5% random noise

Prediction Models and Results

Baseline 1: Simple rpart on raw variables

```

library(rpart)

library(rpart.plot)

tree_baseline <- rpart(

  Social_Media_Burnout ~ Age +

  Avg_Daily_Usage_Hours +

  Sleep_Hours_Per_Night +

  Mental_Health_Score +

  Conflicts_Over_Social_Media +

  Addicted_Score,

  data = train,

  method = "class",

  control = rpart.control(cp = 0.01, minbucket = 30)

)

rpart.plot(tree_baseline)

pred_train_tree <- predict(tree_baseline, type = "class")

train_acc_tree <- mean(pred_train_tree == train$Social_Media_Burnout)

pred_test_tree <- predict(tree_baseline, newdata = test, type = "class")

test_acc_tree <- mean(pred_test_tree == test$Social_Media_Burnout)

cm_tree <- table(Predicted = pred_test_tree,

                 Actual = test$Social_Media_Burnout)

cm_tree

```

-> This tree only sees raw hours, raw sleep, and raw mental health scores. It may notice that “more usage” and “less sleep” relate to higher burnout, but it does not know about the thresholds at 7 hours of sleep, 4 hours of use, or 6 on the mental score.

Baseline 2: Multinomial logistic regression on raw variables:

```
library(nnet)

mlogit_baseline <- multinom(

  Social_Media_Burnout ~

  Avg_Daily_Usage_Hours +

  Sleep_Hours_Per_Night +

  Conflicts_Over_Social_Media +

  Mental_Health_Score +

  Academic_Level +

  Gender,

  data = train

)

pred_train_mlogit <- predict(mlogit_baseline, type = "class")

train_acc_mlogit <- mean(pred_train_mlogit == train$Social_Media_Burnout)

pred_test_mlogit <- predict(mlogit_baseline, newdata = test, type = "class")

test_acc_mlogit <- mean(pred_test_mlogit == test$Social_Media_Burnout)

cm_mlogit <- table(Predicted = pred_test_mlogit,

                   Actual   = test$Social_Media_Burnout)

cm_mlogit
```

This is an “off-the-shelf” model: it’s standard, simple to run, and uses only the raw variables. It assumes a relatively smooth relationship between each raw feature and

the probability of each burnout class. It **cannot explicitly represent** all the engineered structure (debt, thresholds, platform bumps) I used for the label. In practice, its test accuracy is again **reasonable but clearly not optimal**.

Designed solution: Random forest with engineered features

```
train_eng <- add_engineered_features(train)
```

```
test_eng <- add_engineered_features(test)
```

```
library(randomForest)
```

```
set.seed(2025)
```

```
rf_model <- randomForest(
```

```
  Social_Media_Burnout ~
```

```
  Sleep_Debt + Usage_Over_4 +
```

```
  Ratio_Excess + Mental_Struggle +
```

```
  Conflict_Intensity +
```

```
  Academic_Level + Relationship_Status + Most_Used_Platform,
```

```
  data = train_eng,
```

```
  ntree = 500,
```

```
  mtry = 4,
```

```
  importance = TRUE
```

```
)
```

```
pred_train_rf <- predict(rf_model, type = "class")
```

```
train_acc_rf <- mean(pred_train_rf == train_eng$Social_Media_Burnout)
```

```
pred_test_rf <- predict(rf_model, newdata = test_eng, type = "class")
```

```
test_acc_rf <- mean(pred_test_rf == test_eng$Social_Media_Burnout)
```

```
cm_rf <- table(Predicted = pred_test_rf,  
              Actual   = test_eng$Social_Media_Burnout)
```

```
importance(rf_model)
```

```
varImpPlot(rf_model)
```

- Its inputs (Sleep_Debt, Usage_Over_4, Ratio_Excess, Mental_Struggle, Conflict_Intensity, plus categorical factors) are **exactly the building blocks** used to create risk_score and the burnout label.
- Random forests can capture:
 - nonlinearities
 - interactions (between Academic_Level and Sleep_Debt)
 - and threshold-like behavior very naturally

As a result: **Train accuracy** is high. **Test accuracy** is also high, showing that the model is not just overfitting but actually recovering the underlying rules.

Prediction Models and Accuracy Results

In this section I compare three models:

1. A **simple decision tree (rpart)** using only raw survey variables
2. A **multinomial logistic regression** using raw variables (an “lm-style” baseline)
3. A **random forest with engineered features**, which is my proposed “designed solution”

The goal is to show that:

- **Naive, off-the-shelf models** do reasonably well (around 70–77% accuracy), but
- A model that **reconstructs the hidden rules** and uses a more flexible algorithm can reach **around 96% test accuracy**.

Note: A completely uninformed model would get about one-third accuracy, because the class proportions are roughly balanced:

Baseline 1: Simple decision tree (rpart):

The tree does **much better than random guessing** ($\approx 70\%$ vs $\approx 33\%$).

- It correctly picks up that low sleep, high usage, and poor mental health are associated with higher burnout.
- However, it still confuses the **Moderate** class with both **High** and **Low**, and it cannot fully capture:
 - the **ratio** between usage and sleep,
 - the different **weights by Academic_Level**, and
 - the **relationship-status interactions**.

This is exactly what the homework wants: an off-the-shelf tree does *okay*, but not great.

Baseline 2: Multinomial logistic regression (“lm-style” model)

- Test accuracy jumps to about **76–77%**, which is clearly better than the simple decision tree ($\approx 70\%$).
- Logistic regression captures **smooth, global trends**: more usage & conflicts and less sleep & mental health push you toward the High burnout category.
- But it still assumes **linear relationships** in the raw scale and **does not model thresholds** like “only hours above 4 matter” or “only ratio above 0.8 matters”.
- It also cannot directly express the **piecewise rules by Academic_Level** and the **relationship-specific conflict effects**.

So multinomial logistic regression is a **stronger baseline**, but still a relatively “naive” off-the-shelf method in the context of the hidden rule system.

Designed Solution: Random forest with engineered features

The random forest reaches **about 90.6% accuracy on the training set** and an even higher **95.9% on the test set**.

The slight improvement on the test set (95.9% vs 90.6%) is plausible given random sampling and the added 5% label noise; the model is not perfectly deterministic because the target itself is noisy.

The confusion matrix is **almost perfectly diagonal**:

- Very few Low students are predicted as High (and vice versa)
- Most errors happen between **Moderate** and the neighboring classes, which is expected because the quantile-based cut points for Moderate sit in the middle of the risk distribution

Why the Designed Method Outperforms Naive Approaches

Quantitative comparison

Here is a side-by-side summary of test accuracy:

Model	Features used	Test accuracy
Decision tree (rpart)	Raw survey variables only	0.704
Multinomial logistic regression	Raw survey variables only	0.768
Random forest + engineered features	Hidden-rule-aligned features	0.959

Key points:

- Both off-the-shelf baselines do **much better than chance** (~ 0.33), but they plateau around **70–77%**.
- Once I **recreate the hidden structure** (Sleep_Debt, Usage_Over_4, Ratio_Excess, Mental_Struggle, Conflict_Intensity) and feed them into a flexible model (random forest), accuracy jumps to nearly **96%** on the test set.

This shows that:

- The challenge is **not trivial** (a simple lm-style / rpart model cannot reach 90%).
- But it is also **solvable** by students who: do EDA, engineer the right features and use an appropriate model.

Structural reasons: what naive models miss

The main reason the naive models underperform is **mismatch between their assumptions and how the labels were generated**.

1. Thresholds and hinge functions

- Burnout depends on $\text{pmax}(0, 7 - \text{sleep})$ and $\text{pmax}(0, \text{usage} - 4)$, not simply on raw sleep or usage.
- Naive models (tree/logit on raw features) try to fit **linear or simple splits** on the original scale, so they “smear” the effect instead of capturing the real breakpoints.

2. Ratios and imbalance

- Ratio_Excess encodes how big usage is relative to sleep.

- Logistic regression with separate usage and sleep terms has trouble recovering this “ratio > 0.8” effect.
 - Random forest with the engineered ratio feature can easily learn splits like “if Ratio_Excess > 0.5, predict higher burnout”.
3. **Academic-level-specific weights**
- The code uses different coefficients for High School, Undergraduate, and Graduate students.
 - A single global linear model cannot perfectly represent “different slopes for each academic level” without explicitly adding many interaction terms.
 - The random forest, however, can split first on Academic_Level and then learn level-specific patterns, which mirrors how the rules were written.
4. **Relationship and conflict interactions**
- Undergraduates with "In Relationship" or "Complicated" status get additional risk from conflicts and mental struggle.
 - Baseline models only see raw factors; random forest with engineered features naturally picks up these **interaction patterns** by splitting on Relationship_Status and Conflict_Intensity.
5. **Platform effects**
- Instagram/TikTok raise risk; LinkedIn lowers it. Again, these effects are subtle but consistent, and the forest can combine them with the risk features.

Evidence that the random forest is learning the intended rules

Several outputs support that the random forest is actually exploiting the rule system, not just memorizing the data:

- **Variable importance** places Academic_Level, Mental_Struggle, Conflict_Intensity, Sleep_Debt, and Usage_Over_4 at the top, which are exactly the variables used in the risk_score formula.
- The **confusion matrix** is nearly diagonal, with most remaining errors in the Moderate class, which is where I add label noise and where the quantile cut points are ambiguous.
- The **test accuracy (0.959)** is high but not 1.0, reflecting the intentional 5% label flipping and therefore confirming the model is not overfitting to a perfectly deterministic mapping.

How this satisfies the prediction-challenge design goals

- **Non-trivial:** A single rpart or multinomial model on raw variables **does not reach 90%**.
- **Solvable with insight:** Once students discover features like “hours above 4”, “sleep debt”, and “usage/sleep ratio” and use a more flexible model, **95%+ test accuracy**

becomes achievable.

- **Encourages data fluency, not just button-clicking** To beat the 75% baseline, students must:
 - read the code / description carefully,
 - do EDA and reason about how burnout is constructed,
 - deliberately engineer new features before modeling.

This makes the **Social Media Burnout dataset** a fair, realistic, and educational prediction challenge that clearly distinguishes between naive approaches and a thoughtful, well-designed solution.

