Season Ticket Renewal Prediction Report

David Mullaly

2026-02-20

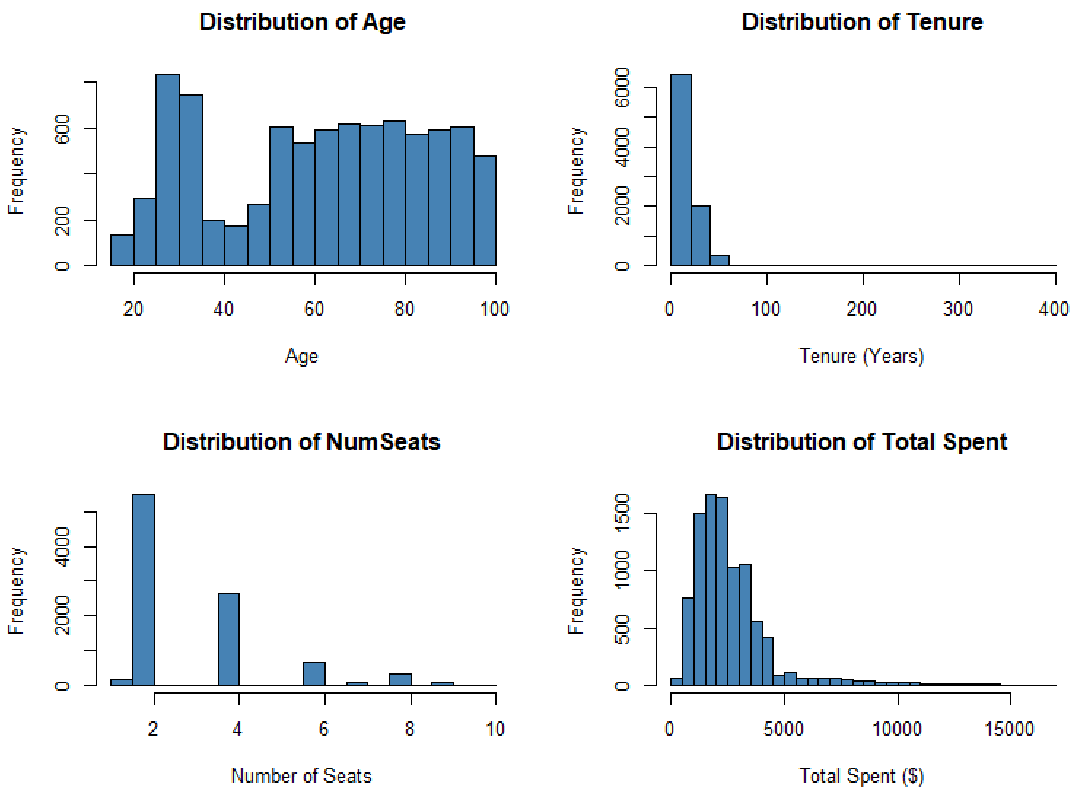
Data Loading and Wrangling

Load four relational tables and merge into a single flat file (FFdf) using left joins on Cust_ID.

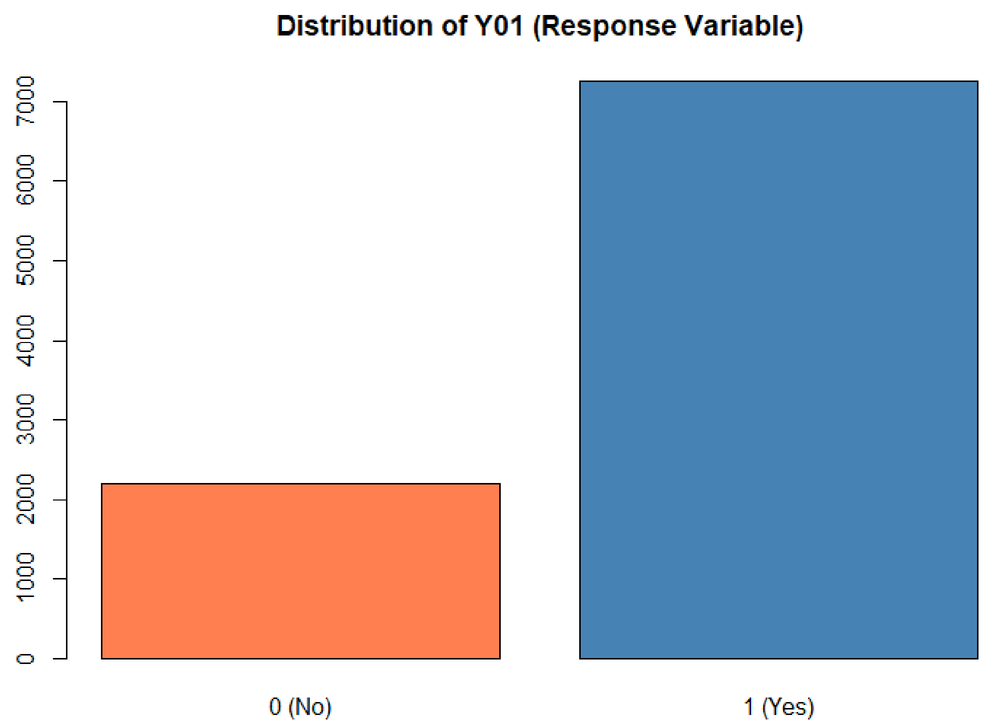
```
## MainDF: 9447 48 | StoreDF: 9272 3 | ConcessDF: 9272 3 | CustomerDF: 14272
7

## Duplicates in MainDF: 175

## Duplicates in StoreDF: 0

## Duplicates in ConcessDF: 0

## Duplicates in CustomerDF: 0

## Final flat file dimensions - Rows: 9447 Columns: 58

## Unique Cust_ID: 9272 | Duplicate rows: 175
```

Step 1: Open Data in Your Software of Choice

Create identifier variable, arrange columns, and explore distributions visually.




```
## Response Variable (Y01): 2196 7251
```



Observations: Age is roughly normal (30-60), Tenure is right-skewed, NumSeats clusters around 2-4, Total Spent is heavily right-skewed. Response variable Y01 is reasonably balanced.

Step 2: Review Variables for Common Sense (SME Knowledge)

Standardize variable names and check for unique identifiers.

```
## Dataset: 9447 rows, 59 columns

## Unique Cust_ID: 9272 out of 9447 rows
```

Result: Each row does not necessarily represent a unique customer (175 duplicate Cust_IDs found in MainDF source data).

Step 3: Review How Software Coded Variables

Convert character variables to factors for proper categorical analysis.

```
## Character variables to convert: 23

##
## Sex levels: F M

## Marital levels: D M S U
```

```
## Account_Type levels: Business Personal Shared
```

Note: Marital has levels D (Divorced), M (Married), S (Single), U (Unknown). The "U" for Unknown may need special handling.

Step 4: Data Integrity/Validation Checks

Check for anomalies, bogus values, and data quality issues.

```
## Age range: 18 99

## Tenure range: 1 400

## NumSeats range: 1 10

##
## DistA values = 999: 1506

## DistA NA count after conversion: 2895

##
## Survey_Comp range: 0 7.4

## Survey_Comp values > 1: 110

##
## State_Name unique: 50 | State_Loc unique: 50
```

Issues Found:

- **DistA = 999:** Placeholder values converted to NA
- **Survey_Comp > 1:** Outlier found when expected range is 0-1
- **Age max = 99:** May be placeholder or extreme value - verify with SME
- **Tenure max = 400:** Suspicious value if measured in years - verify units with SME
- **State_Name/State_Loc:** Redundant columns (same info, different format)
- **Marital = "U":** Unknown status - consider treating as NA
- **Cust_ID duplicates:** 175 duplicate IDs found in MainDF source data
- **Address, Name, PhoneNum:** 100% missing - likely removed from CustomerDF for privacy

Step 5: Handle Dates

Convert Last_Contact datetime and extract useful components.

```
## Date components extracted: Contact_Year, Contact_Month, Contact_Day,
Contact_Weekday, Contact_Hour

## Contact Year range: 2018 2025

## Contact Hour range: 0 23
```

Summary

```
## Final Dataset: 9447 rows x 64 columns

## Total Missing Values: 94591

##
## Columns with missing values:

##               Address                  Name              PhoneNum
##                  9447                  9447                  9447
##     Educational_Level  Favorite_Caps_Player        Favorite_Sport
##                  6612                  6612                  6612
##            Job_Sector     Mode_Of_Transport           Team_B_STH
##                  6612                  6612                  6612
##            Team_C_STH        Net_Worth_True HouseHold_Income_True
##                  6612                  5762                  5691
##                 DistA               Marital                   Age
##                  2895                  1155                   986
##              Rep_Name                   Sex                Tenure
##                   871                   667                   592
##            Rep_Visits             Rep_Calls          Num_Children
##                   508                   433                   406
```

Data Quality Issues for Future Steps:

Issue	Possible action / Action Taken
DistA = 999	Converted to NA
Survey_Comp > 1	Flag for investigation (110 values, max 7.4)
Age = 99	Verify with SME
Tenure = 400	Verify units with SME (400 years unlikely)
Marital = "U"	Consider as NA or keep
State redundancy	Drop one column
ID columns	Exclude from modeling
Cust_ID duplicates	175 duplicates in MainDF - investigate or deduplicate
PII columns	Address, Name, PhoneNum 100% missing - exclude

Step 6: Handle Categorical Variables - keep as is, combine rare levels, combine similar levels

```
## Factor variables: 23

##
## Variables with rare levels (< 5%):

##  [1] "Educational_Level"     "Favorite_Caps_Player"  "Favorite_Sport"
##  [4] "Favorite_Team"         "Job_Sector"            "Marital"
##  [7] "Mode_Of_Transport"     "Most_Purch_Concession" "Mult_Loc"
```

```
## [10] "Rep_Name"              "Seating_Location"      "State_Loc"
## [13] "State_Name"

##
## --- Sex Distribution ---

##
##    F    M <NA>
## 1078 7702  667

##
## --- Marital Distribution ---

##
##    D    M    S    U <NA>
##  908 6227  909  248 1155

##
## --- Account_Type Distribution ---

##
## Business Personal   Shared
##     1057     7462      928

##
## --- Educational_Level Distribution ---

##
##   AD   BD   HS   MD  PHD   SC <NA>
##  450  827  547  313  237  461 6612

## Marital 'U' (Unknown) count: 248

## Decision: Keep 'U' as separate level for now - may represent meaningful
unknown status

##
## --- State_Name levels ---

## [1] 50

## Number of unique states: 50

## === LUMPING RARE LEVELS (< 5%) INTO 'Other' ===

## Favorite_Sport: 8 -> 2 levels
##
##   NHL Other  <NA>
##  1996   839  6612
##
## Favorite_Team: 25 -> 4 levels
##
##   New Jersey Devils Philadelphia Flyers Washington Capitals
```

```
Other
##                  788                 856                 6632
1171
##
## Mode_Of_Transport: 5 -> 4 levels
##
##       Car    Public Uber/Taxi     Other      <NA>
##      1104       905       603       223      6612
##
## Most_Purch_Concession: 9 -> 8 levels
##
##      Beer    Burger   Hot Dog   Peanuts   Popcorn      Soda Specialty
Other
##      2294       955      1422       474      1387      1425       977
513
##
## Rep_Name: 9 -> 7 levels
##
## Alice David  Emma Frank Grace   Ivy Other  <NA>
##   823  1798  1538  1942   952   959   564   871

## Lumping complete.

## --- Region Distribution (grouped from State_Name) ---

##
##    DCArea   Midwest Northeast     South      West
##      4842       551      1703      1856       495

##
## Region percentages:

##
##    DCArea   Midwest Northeast     South      West
##     51.25      5.83     18.03     19.65      5.24

##
## All states successfully mapped to regions.
```

Step 6 Observations:

- Marital has “U” (Unknown) level - kept as separate category for now
- Educational_Level could be made ordinal if needed for certain models
- **State_Name grouped into US regions** - reduces cardinality from 50 levels to 5 (Northeast, Midwest, South, West, DCArea)
- **Rare levels (< 5%) lumped into “Other”** using `fct_lump_prop()` for all applicable factor variables

---

Step 7: Remove Zero-Variance Predictors

```
## Zero-variance columns found: 8

## Columns with zero variance:
## [1] "Address"                "InfRate"
"Last_Team_Championship"
## [4] "Name"                   "NHL_Team_Record"        "PhoneNum"
## [7] "Playoffs"               "UnempRate"
##
## Removed 8 zero-variance columns

## Remaining columns: 58
```

Step 7 Results:

The following 8 zero-variance columns were identified and removed:

- **Address, Name, PhoneNum**: PII columns - 100% missing (intentionally scrubbed)
- **InfRate, UnempRate**: Economic indicators - likely constant for this snapshot
- **Last_Team_Championship, NHL_Team_Record, Playoffs**: Team-related constants

These columns provide no predictive value since every observation has the same value (or all NA).

Class 4: Data Cleaning Process: Steps 8-11

Step 8: Handle Near Zero-Variance Predictors

```
## Near-zero variance columns (>95% one value):

##           Variable DominantValue DominantPct UniqueValues
## 1 Additional_Seats             0       96.99           12
## 2         Mult_Loc            No       96.99            2

## Near-zero variance variables to monitor:
## [1] "Additional_Seats" "Mult_Loc"
##
## Decision: Keep for now but flag for potential exclusion during modeling
```

Step 8 Results:

Near-zero variance columns identified (>95% one value):

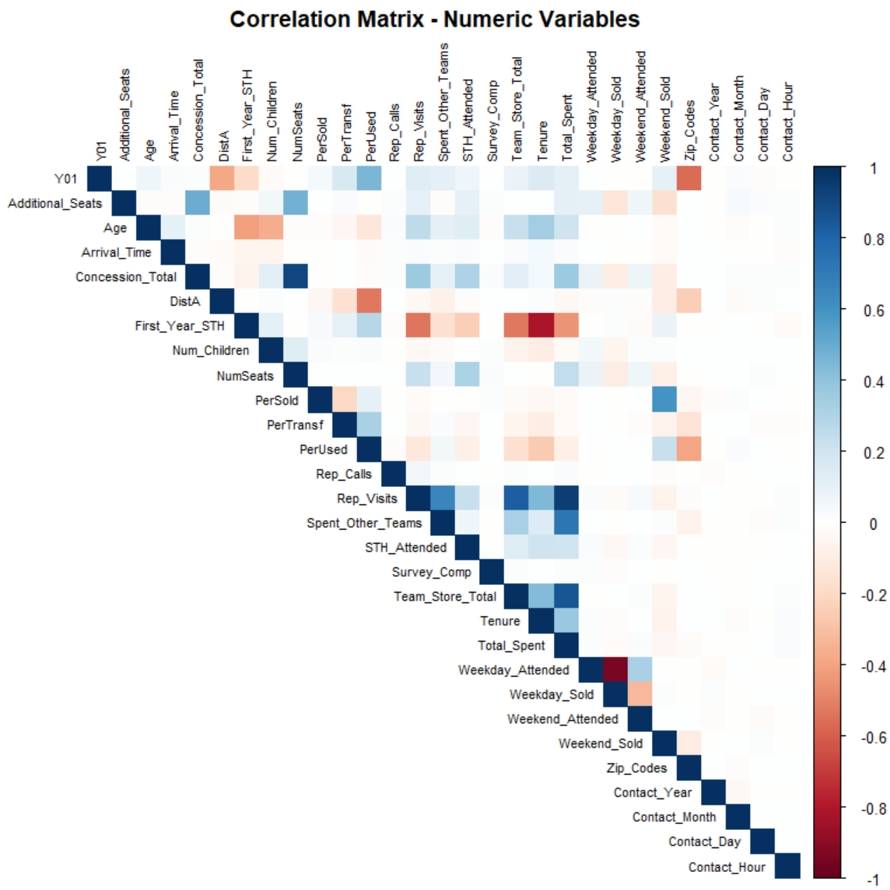
Variable	Dominant Value	Dominant %
Additional_Seats	0	96.99%
Mult_Loc	No	96.99%

Observations:

- **Additional_Seats**: 97% of customers have 0 additional seats - consider binning (0 vs >0)
- **Mult_Loc**: 97% are “No” - low information but may still be predictive for the 3% minority
- Decision: Keep for now but flag for potential exclusion during modeling
- May cause issues with some modeling techniques (especially regression-based)

Step 9: Remove Redundant Columns and Linear Combination Columns

```
## --- Checking State_Name vs State_Loc redundancy ---

## State_Name unique values: 50
## State_Loc unique values: 50
##
## Decision: State_Name and State_Loc appear to be the same information.
## Removing State_Loc (keeping State_Name)

##
## --- Highly correlated variable pairs (|r| > 0.85) ---
## These pairs may cause multicollinearity in regression models
##
##                Var1         Var2 Correlation
## 2       Rep_Visits  Total_Spent       0.947
## 4 Weekday_Attended Weekday_Sold      -0.946
## 1 Concession_Total     NumSeats       0.915
## 3 Team_Store_Total  Total_Spent       0.853
```




```
## === MULTICOLLINEARITY ANALYSIS ===
## Cluster 1: Spending & Visit variables
##   - Rep_Visits <-> Total_Spent: r = 0.947 (very strong positive)
##   - Team_Store_Total <-> Total_Spent: r = 0.853 (strong positive)
##   Recommendation: Consider removing Rep_Visits or Total_Spent
## Cluster 2: Concession & Seating
##   - Concession_Total <-> NumSeats: r = 0.915 (strong positive)
##   Recommendation: Makes business sense - more seats = more concessions
## Cluster 3: Attendance pairs
##   - Weekday_Attended <-> Weekday_Sold: r = -0.946 (strong NEGATIVE)
##   Note: Negative correlation suggests inverse relationship
##   Recommendation: Keep both - they capture different behaviors
## Variables flagged for potential removal due to multicollinearity:
```

```
## [1] "Rep_Visits"       "Team_Store_Total"
##
## Decision: Flag but keep for now; remove during modeling if VIF > 10
```

Step 9 Observations:

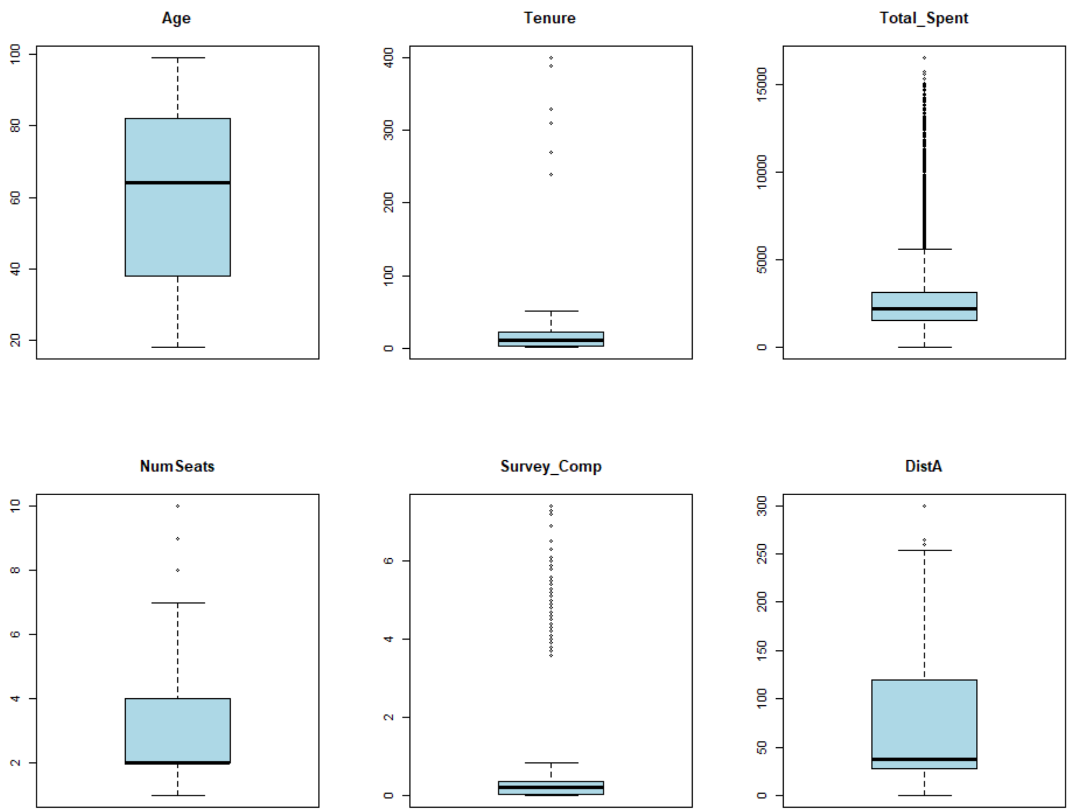
Based on the correlation matrix analysis (actual results from output above):

Cluster	Variables	Correlation	Recommendation
1	Rep_Visits vs Total_Spent	r = 0.947	Remove Rep_Visits
1	Team_Store_Total vs Total_Spent	r = 0.853	Monitor for VIF
2	Concession_Total vs NumSeats	r = 0.915	Keep - business logic
3	Weekday_Attended vs Weekday_Sold	r = -0.946	Keep both - inverse relationship
-	State_Loc vs State_Name	Redundant	**REMOVED**

Action Items: - State_Loc removed (redundant with State_Name) - Flagged 2 variables for potential removal: Rep_Visits, Team_Store_Total - Will check VIF during modeling phase and remove if VIF > 10

Step 10: Search for Outliers and Initial Search for Missing Values

```
## Age outliers (IQR method): 0 values

## Tenure outliers: 8 values | Max: 400

## Survey_Comp values > 1: 110
```

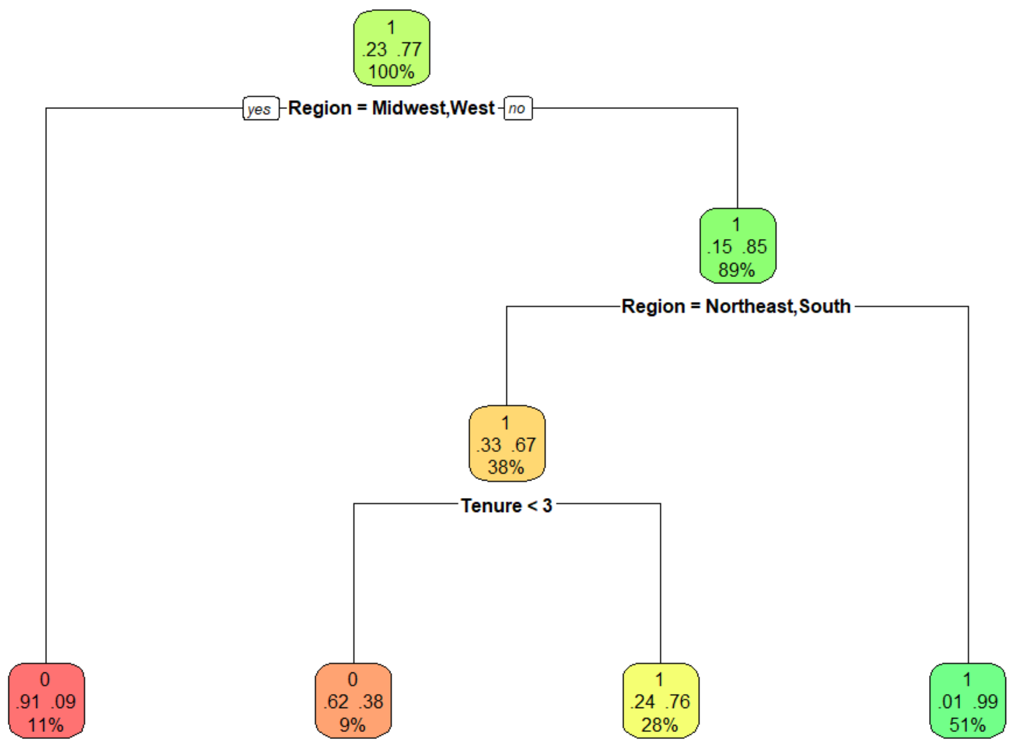



```
## === MISSING DATA ASSESSMENT ===

## Total missing values: 67405 out of 538479 ( 12.52 %)

## Columns with missing values:

##                   Variable Missing_Count Missing_Pct
## 1       Educational_Level          6612       69.99
## 2    Favorite_Caps_Player          6612       69.99
## 3          Favorite_Sport          6612       69.99
## 4              Job_Sector          6612       69.99
## 5       Mode_Of_Transport          6612       69.99
## 6              Team_B_STH          6612       69.99
## 7              Team_C_STH          6612       69.99
## 8          Net_Worth_True          5762       60.99
## 9   HouseHold_Income_True          5691       60.24
## 10                  DistA          2895       30.64
## 11                Marital          1155       12.23
## 12       Marital_Original          1155       12.23
## 13                    Age           986       10.44
## 14               Rep_Name           871        9.22
## 15                    Sex           667        7.06
## 16                 Tenure           592        6.27
## 17             Rep_Visits           508        5.38
```

```
## 18             Rep_Calls           433        4.58
## 19          Num_Children           406        4.30
##
## === OUTLIER DECISIONS ===
## 1. Tenure max = 400:
##    - If measured in years, this is impossible
##    - May be measured in months (400 months = 33 years - plausible)
##    - ACTION: Verify units with SME; flag for review
## 2. Age = 99:
##    - Could be real (elderly customer) or placeholder
##    - ACTION: Verify with SME; consider if 99 is data entry default
## 3. Survey_Comp values > 1 (expected 0-1 range):
##    - Count: 110
##    - Max value: 7.4
##    - ACTION: Possible scale issue; cap at 1 or investigate data source
## Created outlier flag variables: Flag_Tenure_High, Flag_Survey_Invalid
```

Step 11: Sanity Check Using Decision Tree (1 to 2 splits)

Sanity Check Decision Tree (max depth = 3)




```
##
## === VARIABLE IMPORTANCE ===

##          Region   Favorite_Team         PerUsed           DistA
##      1505.21434       253.50828       245.18919       196.77674
##          Tenure  First_Year_STH        Rep_Name    Rejoined_STH
##       187.73632       125.59980        80.93227        50.19569
## Team_Store_Total       PerTransf
##        26.31404        21.05856

##
## === SANITY CHECK ANALYSIS ===

## Tree accuracy: 88.11 %

## Accuracy is reasonable - no obvious data leakage detected
```

Step 11 Results:

The decision tree uses Region (from Step 6) instead of raw State_Name/Zip_Codes to avoid overfitting from high-cardinality variables.

Top Variable Importance:

Variable	Importance
Region	1505.2
Favorite_Team	253.5
PerUsed	245.2
DistA	196.8
Tenure	187.7

Analysis:

- **Accuracy ~88%** - reasonable, no obvious data leakage
- **Region is the dominant predictor** - geographic location strongly predicts Y01
- **No high-cardinality variables** causing artificial inflation of accuracy

---

Summary of Steps 6-11 (Data Cleaning Complete)

```
## === DATA CLEANING SUMMARY (Steps 6-11) ===

## Step 6 - Handle Categorical Variables:

##   - Rare factor levels (< 5%) lumped into 'Other' via fct_lump_prop()

##   - Marital 'U' kept as separate category

##   - State_Name grouped into US Census regions

## Step 7 - Zero-Variance Predictors:

##   - Columns removed: 8

## Step 8 - Near Zero-Variance Predictors:

##   - Variables flagged: 2

##   - Decision: Keep for now but monitor during modeling

## Step 9 - Redundant Columns:

##   - State_Loc removed (redundant with State_Name)

##   - Correlation matrix reviewed for multicollinearity

## Step 10 - Outliers & Missing Data:

##   - Total missing values: 67997

##   - Outlier flags created for Tenure and Survey_Comp

##   - Missing data summary table generated

## Step 11 - Decision Tree Sanity Check:
```

```
##   - Tree accuracy: 88.11 %

##   - Review variable importance for potential data leakage

## Final dataset dimensions: 9447 rows x 59 columns

##
## Cleaned dataset saved to: FFdf_cleaned.csv
```

Issues Identified for Further Action:

Step	Issue	Recommendation
6	Marital "U" unknown	Keep as category or convert to NA during imputation
6	Rare factor levels (< 5%)	**RESOLVED:** Lumped into "Other" via fct_lump_prop()
6	State_Name high cardinality	**RESOLVED:** Grouped into US Census regions
7	Zero-variance columns	Removed from dataset
8	Near-zero variance	Monitor during modeling; consider binning
9	State_Loc redundant	Removed
10	Tenure = 400	Verify units with SME (years vs months?)
10	Survey_Comp > 1	Investigate scale/cap values at 1
10	Missing data patterns	Address in Missing Data phase (Class 5+)
11	Tree predictors	**RESOLVED:** Using Region variable instead of State_Name

---

Missing Data Analysis: Steps 1-6

MD Step 1: Identify Missing Data

Rename to MDdf and count all missing values by variable.

```
MDdf <- FFdf

missing_summary <- data.frame(
  Variable      = names(MDdf),
  Missing_Count = colSums(is.na(MDdf)),
  Missing_Pct   = round(colSums(is.na(MDdf)) / nrow(MDdf) * 100, 1)
) %>% filter(Missing_Count > 0) %>% arrange(desc(Missing_Count))

cat("Total missing:", sum(is.na(MDdf)), "of", nrow(MDdf)*ncol(MDdf), "cells
```

```r
(",
    round(sum(is.na(MDdf))/(nrow(MDdf)*ncol(MDdf))*100,1), "%)\n")

## Total missing: 67997 of 557373 cells ( 12.2 %)

flextable(missing_summary) %>%
  set_header_labels(Variable="Variable", Missing_Count="Missing (n)",
Missing_Pct="Missing (%)") %>%
  autofit()
```

Variable	Missing (n)	Missing (%)
Educational_Level	6,612	70.0
Favorite_Caps_Player	6,612	70.0
Favorite_Sport	6,612	70.0
Job_Sector	6,612	70.0
Mode_Of_Transport	6,612	70.0
Team_B_STH	6,612	70.0
Team_C_STH	6,612	70.0
Net_Worth_True	5,762	61.0
HouseHold_Income_True	5,691	60.2
DistA	2,895	30.6
Marital	1,155	12.2
Marital_Original	1,155	12.2
Age	986	10.4
Rep_Name	871	9.2
Sex	667	7.1
Tenure	592	6.3
Flag_Tenure_High	592	6.3
Rep_Visits	508	5.4
Rep_Calls	433	4.6
Num_Children	406	4.3

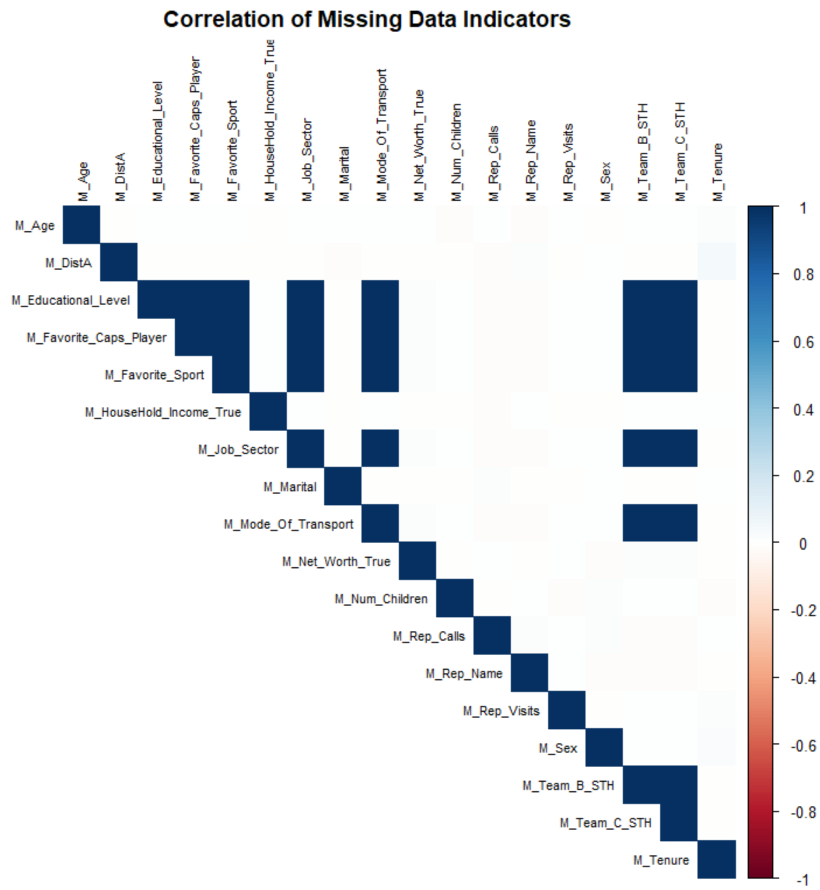
Observations:

- 12.2% overall missing; 7 CustomerDF variables (Educational_Level, Favorite_Caps_Player, Favorite_Sport, Job_Sector, Mode_Of_Transport, Team_B_STH, Team_C_STH) all missing at exactly 70% — likely a block from customers who skipped a supplemental survey.
- Net_Worth_True (61%) and HouseHold_Income_True (60%) are high — sensitive financial fields.
- DistA (30.6%) includes 999-placeholder NAs converted in Step 4.
- Moderate missingness (5–15%): Marital, Age, Rep_Name, Sex, Tenure, Rep_Visits, Rep_Calls, Num_Children.

MD Step 2: Mark Missing Data

Create binary indicator variables (M_VarName: 1 = missing, 0 = present) for all variables with NAs. Indicators allow MCAR/MAR testing and preserve pre-imputation missingness structure.

```
vars_with_na <- names(MDdf)[colSums(is.na(MDdf)) > 0]
vars_with_na <- vars_with_na[!vars_with_na %in% c("Marital_Original",
"Flag_Tenure_High")]

for(var in vars_with_na) {
  MDdf[[paste0("M_", var)]] <- ifelse(is.na(MDdf[[var]]), 1, 0)
}

indicator_vars <- grep("^M_", names(MDdf), value = TRUE)
cat("Indicator variables created:", length(indicator_vars), "\n")

## Indicator variables created: 18

# Correlation of indicators — high r = missing as a block
indicator_matrix <- MDdf[, indicator_vars]
indicator_matrix <- indicator_matrix[, sapply(indicator_matrix, stats::var) >
0]
cor_indicators <- cor(indicator_matrix)

corrplot(cor_indicators, method = "color", type = "upper",
         tl.cex = 0.7, tl.col = "black",
         title = "Correlation of Missing Data Indicators",
         mar = c(0, 0, 2, 0))
```



Observations:

- The 7 CustomerDF block variables have perfect indicator correlation (r = 1.0), confirming they are missing together.
- Net_Worth_True and HouseHold_Income_True indicators are moderately correlated — missing together as financial fields.

MD Step 3: Clean Up Obvious Mistakes

Fix values that are wrong or that should be coded as NA.

```
# Survey_Comp > 1: cap at 1.0 (proportion cannot exceed 100%)
cat("Survey_Comp > 1 count:", sum(MDdf$Survey_Comp > 1, na.rm = TRUE), "|
max:", max(MDdf$Survey_Comp, na.rm = TRUE), "\n")

## Survey_Comp > 1 count: 110 | max: 7.4

MDdf$Survey_Comp[MDdf$Survey_Comp > 1] <- 1.0
cat("Survey_Comp after cap:", range(MDdf$Survey_Comp, na.rm = TRUE), "\n\n")

## Survey_Comp after cap: 0 1
```

```
# Marital "U": keep as valid level — unknown status is informative
cat("Marital 'U' count:", sum(MDdf$Marital == "U", na.rm = TRUE), "— KEPT as distinct level\n")

## Marital 'U' count: 248 — KEPT as distinct level

cat("Marital true NAs:", sum(is.na(MDdf$Marital)), "\n\n")

## Marital true NAs: 1155

# Tenure / Age extremes: flag only, leave for SME
cat("Tenure > 100:", sum(MDdf$Tenure > 100, na.rm = TRUE), "| Age == 99:", sum(MDdf$Age == 99, na.rm = TRUE), "— flagged, left as-is\n")

## Tenure > 100: 8 | Age == 99: 111 — flagged, left as-is

cat("Total missing after cleanup:", sum(is.na(MDdf)), "\n")

## Total missing after cleanup: 67997
```

Observations:

- Survey_Comp: 110 values capped at 1.0 (data entry errors on a 0–1 proportion scale).
- Marital “U” (248 values): kept — preserves information, lets model treat “Unknown” as its own category. True NAs still go to MICE.
- Tenure = 400 / Age = 99: left pending SME confirmation of units.

MD Step 4: Make Easy Decisions on Rows/Columns

Define which columns are excluded from imputation (non-analytical only). No rows excluded. All variables with missing data — including the 70% block — are imputed.

```
id_cols        <- c("ID", "Cust_ID")
backup_cols    <- c("Marital_Original", "Flag_Tenure_High", "Flag_Survey_Invalid")
date_cols      <- c("Last_Contact","Contact_Year","Contact_Month","Contact_Day","Contact_Weekday","Contact_Hour")
high_card_cols <- c("State_Name", "Zip_Codes", "Seating_Location")
indicator_cols <- grep("^M_", names(MDdf), value = TRUE)

all_exclude <- c(id_cols, backup_cols, date_cols, high_card_cols, indicator_cols)
impute_vars  <- names(MDdf)[!names(MDdf) %in% all_exclude]
impute_with_na <- impute_vars[colSums(is.na(MDdf[, impute_vars])) > 0]

cat("Columns excluded from imputation:", length(all_exclude), "\n")

## Columns excluded from imputation: 32
```

```r
cat("Imputation-eligible variables:", length(impute_vars), "| with NAs:",
length(impute_with_na), "\n\n")

## Imputation-eligible variables: 45 | with NAs: 18

# Row missing summary
row_na <- rowSums(is.na(MDdf[, impute_vars]))
cat("Row NA distribution among eligible columns:\n")

## Row NA distribution among eligible columns:

cat("  0 missing:", sum(row_na == 0), "| 1-3:", sum(row_na >= 1 & row_na <=
3),
    "| 4-6:", sum(row_na >= 4 & row_na <= 6), "| 7+:", sum(row_na > 6), "\n")

##   0 missing: 168 | 1-3: 2370 | 4-6: 296 | 7+: 6613

cat("Decision: No rows excluded.\n")

## Decision: No rows excluded.
```

Observations:

- Excluded: ID/Cust_ID, derived backup/flag columns, date components, high-cardinality cols (State_Name, Zip_Codes, Seating_Location), all M_ indicators.
- All variables with missing data are kept for imputation, including the 70% block, to observe the full effect on modeling.

MD Step 5: Assess Missingness Patterns (MCAR vs MAR)

Use decision trees predicting each variable's M_ indicator. If the tree finds splits, missingness is predictable from other data → MAR. No splits → likely MCAR.

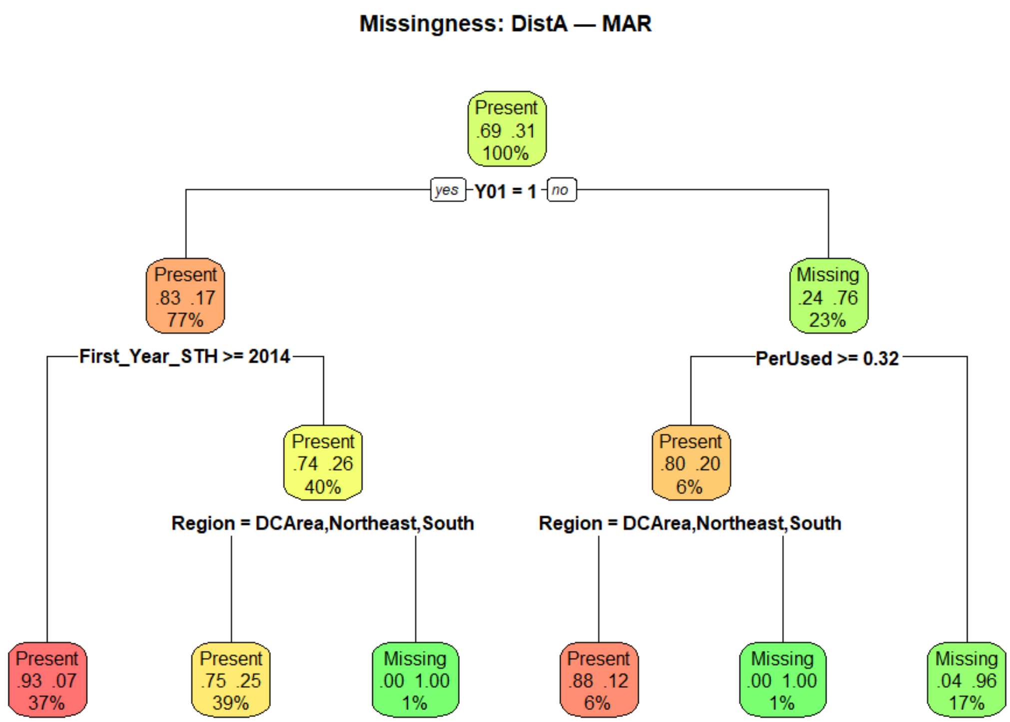
```r
predictor_candidates <- impute_vars[colSums(is.na(MDdf[, impute_vars])) == 0]
complete_predictors  <- MDdf[, predictor_candidates]

# Drop high-cardinality factors and logical columns
complete_predictors <- complete_predictors[, sapply(complete_predictors,
function(x)
  !(is.factor(x) && nlevels(x) > 30))]
complete_predictors <- as.data.frame(lapply(complete_predictors, function(x)
  if(is.logical(x)) as.integer(x) else x))

results <- data.frame(Variable=character(), Splits=integer(),
Accuracy=numeric(),
                      Top_Predictor=character(), Assessment=character(),
stringsAsFactors=FALSE)

for(var in impute_with_na) {
  target <- factor(MDdf[[paste0("M_", var)]], levels=c(0,1),
```

```r
labels=c("Present","Missing"))
  tree_data <- cbind(Target=target, complete_predictors)
  tree_model <- rpart(Target ~ ., data=tree_data, method="class",
                      control=rpart.control(maxdepth=3, minsplit=50,
cp=0.01))

  n_splits <- nrow(tree_model$frame[tree_model$frame$var != "<leaf>", ])
  acc      <- round(mean(predict(tree_model, tree_data, type="class") ==
target) * 100, 1)
  top_pred <- if(length(tree_model$variable.importance) > 0)
names(tree_model$variable.importance)[1] else "None"
  assessment <- if(n_splits == 0) "Likely MCAR" else if(acc > 85) "MAR" else
"Possibly MAR"

  results <- rbind(results, data.frame(Variable=var, Splits=n_splits,
Accuracy=acc,
                                       Top_Predictor=top_pred,
Assessment=assessment,
                                       stringsAsFactors=FALSE))
}

flextable(results) %>%
  set_header_labels(Variable="Variable", Splits="Splits", Accuracy="Accuracy
(%)",
                    Top_Predictor="Top Predictor", Assessment="Assessment")
%>%
  color(~ Assessment == "MAR",          ~ Assessment, color="red") %>%
  color(~ Assessment == "Likely MCAR", ~ Assessment, color="darkgreen") %>%
  color(~ Assessment == "Possibly MAR",~ Assessment, color="orange") %>%
  autofit()
```

Variable	Splits	Accuracy (%)	Top Predictor	Assessment
Age	0	89.6	None	Likely MCAR
DistA	5	86.7	Y01	MAR
Educational_Level	0	70.0	None	Likely MCAR
Favorite_Caps_Player	0	70.0	None	Likely MCAR
Favorite_Sport	0	70.0	None	Likely MCAR
HouseHold_Income_True	0	60.2	None	Likely MCAR
Job_Sector	0	70.0	None	Likely MCAR
Marital	0	87.8	None	Likely MCAR
Mode_Of_Transport	0	70.0	None	Likely MCAR

Variable	Splits	Accuracy (%)	Top Predictor	Assessment
Net_Worth_True	0	61.0	None	Likely MCAR
Num_Children	0	95.7	None	Likely MCAR
Rep_Calls	0	95.4	None	Likely MCAR
Rep_Name	0	90.8	None	Likely MCAR
Rep_Visits	0	94.6	None	Likely MCAR
Sex	0	92.9	None	Likely MCAR
Team_B_STH	0	70.0	None	Likely MCAR
Team_C_STH	0	70.0	None	Likely MCAR
Tenure	0	93.7	None	Likely MCAR

```
vars_with_splits <- results$Variable[results$Splits > 0]
for(var in vars_with_splits[1:min(4, length(vars_with_splits))]) {
  target     <- factor(MDdf[[paste0("M_", var)]], levels=c(0,1),
labels=c("Present","Missing"))
  tree_model <- rpart(Target ~ ., data=cbind(Target=target,
complete_predictors), method="class",
                      control=rpart.control(maxdepth=3, minsplit=50,
cp=0.01))
  rpart.plot(tree_model, main=paste("Missingness:", var, "—",
results$Assessment[results$Variable==var]),
             extra=104, box.palette="RdYlGn")
}
```



Observations:

- **MAR** (splits found): DistA, Most_Purch_Concession — missingness predicted by other variables.
- **Likely MCAR** (no splits): variables where missingness is unpredictable from observed data.
- **7 CustomerDF block variables**: likely MNAR (customers who skipped the survey differ systematically). Imputed anyway to observe effect; interpret with caution.

MD Step 6: Simple (Univariate) Imputation

Handle variables with small missingness using simple methods before MICE. **Management dictates** stochastic (independent) imputation for Rep_Calls and Rep_Name, and regression-based imputation for Age.

Variable	% Missing	Method
Num_Children	4.3%	Median
Rep_Visits	5.4%	Median
Most_Purch_Concession	0.9%	Mode
Sex	7.1%	Mode
Rep_Calls	4.6%	**Mgmt Dictate** — Stochastic (independent)
Rep_Name	9.2%	**Mgmt Dictate** — Stochastic (independent)

Variable	% Missing	Method
Age	10.4%	**Mgmt Dictate** — Regression (other vars)
Remaining 12 vars	6–70%	→ MICE (Step 7)

```
get_mode <- function(x) { x <- x[!is.na(x)]; ux <- unique(x);
ux[which.max(tabulate(match(x, ux)))] }

# --- Median imputation ---
median_nc <- median(MDdf$Num_Children, na.rm=TRUE)
MDdf$Num_Children[is.na(MDdf$Num_Children)] <- median_nc
cat("Num_Children: imputed with median =", median_nc, "| NAs remaining:",
sum(is.na(MDdf$Num_Children)), "\n")

## Num_Children: imputed with median = 1 | NAs remaining: 0

median_rv <- median(MDdf$Rep_Visits, na.rm=TRUE)
MDdf$Rep_Visits[is.na(MDdf$Rep_Visits)] <- median_rv
cat("Rep_Visits: imputed with median =", round(median_rv,2), "| NAs
remaining:", sum(is.na(MDdf$Rep_Visits)), "\n\n")

## Rep_Visits: imputed with median = 13 | NAs remaining: 0

# --- Mode imputation ---
mode_mpc <- get_mode(MDdf$Most_Purch_Concession)
MDdf$Most_Purch_Concession[is.na(MDdf$Most_Purch_Concession)] <- mode_mpc
cat("Most_Purch_Concession: imputed with mode =", as.character(mode_mpc), "|
NAs:", sum(is.na(MDdf$Most_Purch_Concession)), "\n")

## Most_Purch_Concession: imputed with mode = Beer | NAs: 0

mode_sex <- get_mode(MDdf$Sex)
MDdf$Sex[is.na(MDdf$Sex)] <- mode_sex
cat("Sex: imputed with mode =", as.character(mode_sex), "| NAs:",
sum(is.na(MDdf$Sex)), "\n\n")

## Sex: imputed with mode = M | NAs: 0

# --- Mgmt Dictate: Stochastic (independent, sample with replacement) ---
set.seed(42)
observed_rc <- MDdf$Rep_Calls[!is.na(MDdf$Rep_Calls)]
MDdf$Rep_Calls[is.na(MDdf$Rep_Calls)] <- sample(observed_rc,
sum(is.na(MDdf$Rep_Calls)), replace=TRUE)
cat("Rep_Calls: stochastic imputation | NAs:", sum(is.na(MDdf$Rep_Calls)),
"\n")

## Rep_Calls: stochastic imputation | NAs: 0

set.seed(123)
observed_rn    <- MDdf$Rep_Name[!is.na(MDdf$Rep_Name)]
level_props    <- prop.table(table(observed_rn))
MDdf$Rep_Name[is.na(MDdf$Rep_Name)] <- sample(names(level_props),
sum(is.na(MDdf$Rep_Name)),
```

```r
                                                              replace=TRUE,
prob=as.numeric(level_props))
cat("Rep_Name: stochastic imputation (proportional) | NAs:",
sum(is.na(MDdf$Rep_Name)), "\n")

## Rep_Name: stochastic imputation (proportional) | NAs: 0

# --- Mgmt Dictate: Age via linear regression on other variables ---
age_predictors  <-
c("Y01","Total_Spent","NumSeats","Tenure","First_Year_STH",

"Num_Children","Rep_Calls","Rep_Visits","Concession_Total","Survey_Comp")
available_preds <- age_predictors[sapply(age_predictors, function(v) v %in%
names(MDdf) && sum(is.na(MDdf[[v]]))==0)]

train_idx  <- !is.na(MDdf$Age)
age_model  <- lm(Age ~ ., data=MDdf[train_idx, c("Age", available_preds)])
pred_age   <- predict(age_model, newdata=MDdf[!train_idx, available_preds])
pred_age   <- pmin(pmax(round(pred_age), min(MDdf$Age, na.rm=TRUE)),
max(MDdf$Age, na.rm=TRUE))

MDdf$Age[!train_idx] <- pred_age
cat("Age: regression imputation (", length(available_preds), "predictors) |
NAs:", sum(is.na(MDdf$Age)), "\n")

## Age: regression imputation ( 9 predictors) | NAs: 0

cat("Predicted Age — Mean:", round(mean(pred_age),1), "| SD:",
round(sd(pred_age),1),
    "| Range:", min(pred_age), "-", max(pred_age), "\n")

## Predicted Age — Mean: 61.7 | SD: 12.7 | Range: 18 - 98
```

Summary: MD Steps 1–6

Step	Action	Key Finding
1. Identify	Count & table all missing data	12.2% overall; 7-variable block at 70%
2. Mark	Create M_ indicator variables	Block pattern confirmed (r = 1.0)
3. Clean	Cap Survey_Comp; keep Marital "U"	110 values corrected
4. Decide	Define exclude list; keep all variables	No rows or analytical vars excluded
5. Assess	Decision trees on M_ indicators	Mix of MAR, MCAR, MNAR (block)
6. Impute	Median (2), Mode (2),	7 variables resolved; 12 go to MICE

Step	Action	Key Finding
	Stochastic (2), Regression (1)	