

Cal Poly Pomona Centerpointe Dataset Generator - Complete Manual

Table of Contents

- 1. [Overview](#)
- 2. [Configuration Parameters](#)
- 3. [Output Dataset Columns](#)
- 4. [Parameter Impact Guide](#)
- 5. [Common Configuration Scenarios](#)
- 6. [Troubleshooting & Validation](#)

Overview

The `CenterpointeDiningDataGenerator` creates realistic synthetic datasets for Cal Poly Pomona's Centerpointe Dining Commons. It models complex university dining operations including academic calendars, student behavior, meal plans, weather effects, campus events, and staffing requirements across six operational roles.

Primary Use Case: Generate training data for machine learning models that predict optimal staffing levels based on operational conditions.

Configuration Parameters

Core Parameters

`random_seed` (int, default: 42)

Controls reproducibility of random number generation.

- **Range:** Any integer
- **Impact:** Same seed produces identical datasets
- **Usage:** Set to different values for ensemble training, same value for consistent results

`start_date` (str, default: '2024-01-01')

Start date for dataset generation in YYYY-MM-DD format.

- **Format:** ISO date string
- **Recommendation:** Align with academic calendar (e.g., start of semester)
- **Impact:** Determines total dataset size and seasonal patterns

end_date (str, default: '2026-12-31')

End date for dataset generation in YYYY-MM-DD format.

- **Format:** ISO date string
- **Recommendation:** 1-4 years of data for robust ML training
- **Impact:** Longer periods capture more seasonal variation and growth trends

facility_name (str, default: 'Centerpointe Dining Commons')

Identifier for the facility being modeled.

- **Usage:** Documentation and output labeling
 - **Impact:** No direct impact on data generation
-

Student Population Parameters

student_population.total_enrollment_base (int, default: 31000)

Base university enrollment for the starting year.

- **Range:** 15,000 - 50,000 (typical university range)
- **CPP Actual:** ~31,000 students
- **Impact:** Primary driver of transaction volume and meal plan participation

student_population.yoy_growth_rate (float, default: 0.022)

Annual enrollment growth rate as decimal.

- **Range:** -0.05 to 0.05 (-5% to +5% annually)
- **Typical:** 0.015 to 0.03 (1.5% to 3% growth)
- **Impact:** Affects long-term transaction trends and capacity planning

student_population.residential_student_ratio (float, default: 0.152)

Percentage of students living on campus.

- **Range:** 0.10 to 0.30 (10% to 30%)
- **CPP Actual:** ~15.2%
- **Impact:** Higher ratios increase meal plan participation significantly

student_population.meal_plan_participation.residential_mandatory_rate (float, default: 1.0)

Percentage of residential students required to have meal plans.

- **Range:** 0.90 to 1.0 (90% to 100%)
- **CPP Policy:** 100% mandatory
- **Impact:** Direct multiplier on guaranteed meal plan revenue

`student_population.meal_plan_participation.commuter_voluntary_rate` (float, default: 0.078)

Percentage of commuter students who voluntarily purchase meal plans.

- **Range:** 0.05 to 0.15 (5% to 15%)
- **Impact:** Affects transaction volume variability and revenue diversity

`student_population.enrollment_seasonal_variation`

Enrollment multipliers by academic period.

- **fall_semester** (default: 1.0): Peak enrollment period
 - **spring_semester** (default: 0.96): Some students don't return
 - **summer_session** (default: 0.32): Limited programs and housing
 - **winter_intersession** (default: 0.08): Minimal campus population
-

Meal Plan System Parameters

`meal_plans.plan_types.[PLAN_NAME].cost_per_semester` (int)

Semester cost in dollars for each meal plan type.

- **CPP 2024-2025 Actual Costs:**
 - Unlimited: \$2,611
 - Block_220: \$2,895
 - Block_180: \$2,781
 - Block_140: \$2,611
 - Suites_Flex: \$1,915

`meal_plans.plan_types.[PLAN_NAME].typical_daily_usage` (float)

Average meals per day when plan is actively used.

- **Unlimited:** 2.4 meals/day
- **Block_220:** 1.83 meals/day (220 swipes ÷ 120 semester days)
- **Block_180:** 1.50 meals/day
- **Block_140:** 1.17 meals/day
- **Suites_Flex:** 0.58 meals/day (have kitchens)

`meal_plans.plan_types.[PLAN_NAME].utilization_rate` (float)

Percentage of purchased meals actually consumed.

- **Range:** 0.70 to 0.95
- **Factors:** Plan type, kitchen access, social commitments
- **Impact:** Lower utilization = more conservative transaction estimates

`meal_plans.plan_types.[PLAN_NAME].student_distribution` (float)

Percentage of meal plan holders with each plan type.

- **Must sum to 1.0** across all plan types
- **CPP Distribution:**
 - Unlimited: 28%
 - Block_220: 24%
 - Block_180: 26%
 - Block_140: 17%
 - Suites_Flex: 5%

Operating Hours Parameters

`operating_hours.academic_year.weekday`

Standard weekday meal periods during academic year.

- **breakfast:** (7.0, 10.0) = 7:00 AM to 10:00 AM
- **lunch:** (10.75, 14.5) = 10:45 AM to 2:30 PM
- **dinner:** (17.0, 19.5) = 5:00 PM to 7:30 PM
- **late_night:** (21.0, 23.0) = 9:00 PM to 11:00 PM

`operating_hours.academic_year.weekend`

Weekend schedule with brunch replacing breakfast/lunch.

- **brunch:** (11.0, 15.0) = 11:00 AM to 3:00 PM
- **dinner:** (17.0, 19.5) = 5:00 PM to 7:30 PM
- **late_night:** (21.0, 22.5) = 9:00 PM to 10:30 PM

operating_hours.summer_session

Reduced summer operating hours.

- **weekday:** lunch (11.0, 14.0), dinner (17.0, 19.0)
- **weekend:** lunch (11.5, 14.0), dinner (17.0, 18.5)

operating_hours.break_periods

Minimal operations during academic breaks.

- **weekday:** lunch (11.5, 13.5) only
- **weekend:** 'closed'

Academic Calendar Parameters

academic_calendar.semester_dates

Key dates defining academic periods.

- **fall_start:** (8, 20) = August 20
- **fall_end:** (12, 15) = December 15
- **spring_start:** (1, 15) = January 15
- **spring_end:** (5, 15) = May 15
- **summer_start:** (6, 1) = June 1
- **summer_end:** (8, 15) = August 15

academic_calendar.special_periods.[PERIOD_NAME].multiplier (float)

Traffic multipliers for special academic periods.

- **move_in_week:** 1.28 (+28% traffic from new students exploring)
- **finals_weeks:** 1.16 (+16% traffic from extended hours, stress eating)
- **spring_break:** 0.31 (-69% traffic, most students away)
- **winter_intersession:** 0.09 (-91% traffic, minimal programs)
- **thanksgiving_week:** 0.45 (-55% traffic, travel home)

academic_calendar.weekly_patterns

Day-of-week multipliers reflecting campus activity patterns.

- **monday:** 0.92 (slow start to week)
 - **tuesday:** 1.05 (peak academic day)
 - **wednesday:** 1.08 (peak academic day with U-Hour)
 - **thursday:** 1.03 (busy academic day)
 - **friday:** 0.89 (students leave campus early)
 - **saturday:** 0.71 (weekend pattern)
 - **sunday:** 0.82 (students return, meal prep)
-

Environmental Factors Parameters

`environmental_factors.weather_patterns.seasonal_probabilities`

Weather probability distributions by season [sunny, cloudy, rainy, extreme_heat].

- **winter:** [0.58, 0.27, 0.14, 0.01] (more rain in SoCal winter)
- **spring:** [0.72, 0.19, 0.08, 0.01] (mild season)
- **summer:** [0.79, 0.17, 0.02, 0.02] (hot, dry season)
- **fall:** [0.71, 0.22, 0.06, 0.01] (fire season, clear skies)

`environmental_factors.weather_patterns.weather_impacts`

Traffic multipliers for different weather conditions.

- **sunny:** 1.0 (baseline)
- **cloudy:** 1.023 (+2.3% slight increase)
- **rainy:** 1.147 (+14.7% students stay indoors)
- **extreme_heat:** 0.891 (-10.9% reduced appetite)

`environmental_factors.campus_events.event_calendar.[EVENT_NAME]`

Campus event definitions with probability and impact.

- **regular_day:** probability 0.823, impact 1.0
- **club_fair:** probability 0.025, impact 1.34 (+34% traffic)
- **career_fair:** probability 0.018, impact 1.23 (+23% traffic)
- **graduation:** probability 0.008, impact 1.43 (+43% traffic)
- **parent_weekend:** probability 0.012, impact 1.38 (+38% traffic)
- **campus_construction:** probability 0.020, impact 0.94 (-6% reduced access)

Transaction Pattern Parameters

`transaction_patterns.payment_methods`

Distribution of payment methods reflecting CPP's dual-currency system.

- **meal_swipes:** 0.703 (70.3% primary method)
- **dining_dollars:** 0.198 (19.8% food-only currency)
- **bronco_bucks:** 0.077 (7.7% campus-wide currency)
- **credit_debit:** 0.022 (2.2% external payments, guests)

`transaction_patterns.platform_popularity`

Distribution across Centerpointe's 8 dining platforms.

- **Fusion_Bar:** 0.181 (18.1% most popular, Asian cuisine)
- **Between_Two_Slices:** 0.148 (14.8% sandwich station)
- **Firehouse:** 0.142 (14.2% grilled items)
- **Charred:** 0.134 (13.4% carving station)
- **Gone_Global:** 0.119 (11.9% international foods)
- **Sushi_Bar:** 0.097 (9.7% generates wait times)
- **Salad_Bar:** 0.096 (9.6% healthy options)
- **Sweet_Spot:** 0.083 (8.3% desserts)

`transaction_patterns.meal_period_distribution`

Distribution of daily transactions across meal periods.

- **lunch:** 0.52 (52% peak period)
- **dinner:** 0.27 (27% moderate period)
- **breakfast:** 0.18 (18% light morning crowd)
- **late_night:** 0.03 (3% limited service)

Staffing Model Parameters

`staffing_model.roles.[ROLE_NAME].base_hours_per_period` (float)

Base staffing hours per meal period for each role.

- **FOH_General:** 11.8 hours (customer service, serving, cleaning)
- **FOH_Cashier:** 5.7 hours (transactions, guest relations)
- **Kitchen_Prep:** 15.4 hours (food prep, ingredient processing)
- **Kitchen_Line:** 19.2 hours (active cooking, food assembly)
- **Dish_Room:** 7.9 hours (dishwashing, sanitation)
- **Management:** 3.8 hours (supervision, coordination)

staffing_model.roles.[ROLE_NAME].volume_scaling_factor (float)

How staffing scales with transaction volume.

- **FOH_General:** 1.23 (high volume sensitivity)
- **FOH_Cashier:** 1.14 (moderate volume sensitivity)
- **Kitchen_Prep:** 0.91 (more consistent needs)
- **Kitchen_Line:** 1.02 (moderate scaling)
- **Dish_Room:** 1.17 (scales with total meals)
- **Management:** 0.82 (least variable)

staffing_model.roles.[ROLE_NAME].minimum_coverage_hours (float)

Minimum staffing regardless of volume.

- **Range:** 2.0 to 16.0 hours
- **Purpose:** Ensures basic operational capability
- **Impact:** Sets floor for low-volume periods

staffing_model.labor_costs

Hourly rates for different staff categories.

- **average_hourly_rate:** \$18.75 (blended rate with benefits)
- **student_worker_rate:** \$16.50
- **experienced_staff_rate:** \$21.25
- **management_rate:** \$28.50

staffing_model.scheduling_constraints.student_worker_availability

Availability multipliers for student workers by time period.

- **weekday_morning:** 0.58 (many in class)
 - **weekday_afternoon:** 0.82 (more available)
 - **weekday_evening:** 0.91 (most available)
 - **weekend_day:** 0.97 (highly available)
 - **finals_week:** 0.47 (limited due to studying)
 - **summer_session:** 0.39 (many away from campus)
-

Facility Specifications Parameters

facility_specs.maximum_simultaneous_capacity (int, default: 680)

Maximum number of diners the facility can serve simultaneously.

- **CPP Actual:** 680 seats
- **Impact:** Used for capacity utilization calculations

facility_specs.total_square_footage (int, default: 35000)

Total facility size in square feet.

- **CPP Actual:** 35,000 sq ft
- **Usage:** Documentation and efficiency metrics

facility_specs.number_of_dining_platforms (int, default: 8)

Number of distinct food service platforms.

- **CPP Actual:** 8 platforms
 - **Impact:** Affects platform distribution calculations
-

Output Dataset Columns

Date and Time Information (7 columns)

date (string)

Date in YYYY-MM-DD format.

- **Example:** "2024-10-15"
- **Usage:** Primary time index for analysis

day_of_week (int)

Numeric day of week (0=Monday, 6=Sunday).

- **Range:** 0-6
- **Usage:** Day-of-week pattern analysis

day_name (string)

Full day name.

- **Values:** "Monday" through "Sunday"
- **Usage:** Human-readable day identification

is_weekend (boolean)

True if Saturday or Sunday.

- **Values:** True/False
- **Usage:** Weekend vs weekday analysis

month (int)

Month number.

- **Range:** 1-12
- **Usage:** Seasonal pattern analysis

year (int)

Four-digit year.

- **Example:** 2024
- **Usage:** Year-over-year comparisons

day_of_year (int)

Day of year (1-366).

- **Range:** 1-366
- **Usage:** Annual cyclical patterns

week_of_year (int)

ISO week number.

- **Range:** 1-53
- **Usage:** Weekly trend analysis

academic_period (string)

Current academic period name.

- **Values:** "fall_semester", "spring_semester", "summer_session", "winter_break", "move_in_week", "finals_week", "spring_break", "thanksgiving_week"
- **Usage:** Period-based staffing adjustments

period_type (string)

Type of academic period.

- **Values:** "regular", "special"
- **Usage:** Distinguish normal vs special periods

seasonal_multiplier (float)

Traffic multiplier for current period.

- **Range:** 0.09 to 1.28
- **Usage:** Primary seasonal adjustment factor

period_description (string)

Human-readable period description.

- **Example:** "Regular fall semester", "New students exploring dining options"
 - **Usage:** Documentation and reporting
-

Student Population Data (6 columns)

total_enrollment (int)

Total university enrollment (with growth applied).

- **Range:** 15,000 to 50,000+
- **Usage:** Long-term capacity planning

active_enrollment (int)

Enrollment adjusted for seasonal attendance.

- **Calculation:** $\text{total_enrollment} \times \text{seasonal_factor}$
- **Usage:** Current period planning

residential_students (int)

Number of students living on campus.

- **Calculation:** $\text{active_enrollment} \times \text{residential_ratio}$
- **Usage:** Meal plan participation base

commuter_students (int)

Number of commuter students.

- **Calculation:** $\text{active_enrollment} - \text{residential_students}$
- **Usage:** Voluntary meal plan analysis

total_meal_plan_holders (int)

Total students with meal plans.

- **Calculation:** $(\text{residential} \times \text{mandatory_rate}) + (\text{commuter} \times \text{voluntary_rate})$
- **Usage:** Primary transaction volume driver

enrollment_seasonal_factor (float)

Seasonal attendance multiplier applied.

- **Range:** 0.08 to 1.0
 - **Usage:** Understanding seasonal enrollment impact
-

Environmental Factors (5 columns)

weather (string)

Daily weather condition.

- **Values:** "sunny", "cloudy", "rainy", "extreme_heat"
- **Usage:** Weather impact analysis

weather_impact (float)

Traffic multiplier for weather condition.

- **Range:** 0.891 to 1.147
- **Usage:** Weather-adjusted demand forecasting

campus_event (string)

Type of campus event occurring.

- **Values:** "regular_day", "club_fair", "career_fair", "sports_events", "graduation", "parent_weekend", "conference_hosting", etc.
- **Usage:** Event-based staffing adjustments

event_impact (float)

Traffic multiplier for campus event.

- **Range:** 0.94 to 1.43
- **Usage:** Event-adjusted demand forecasting

event_scheduled (boolean)

Whether event was scheduled vs random.

- **Values:** True/False
 - **Usage:** Distinguish predictable vs unpredictable events
-

Transaction Data (13 columns)

total_transactions (int)

Total daily transactions across all payment methods.

- **Range:** 0 to 8,000+
- **Usage:** Primary staffing predictor variable

guest_transactions (int)

Transactions from non-meal plan holders.

- **Calculation:** ~12% of total transactions
- **Usage:** Guest impact analysis

base_transaction_rate (float)

Transactions per meal plan holder per day.

- **Range:** 0.5 to 3.0
- **Usage:** Meal plan utilization analysis

Payment Method Breakdown (4 columns)

- `meal_swipes` (int): Meal plan swipe transactions
- `dining_dollars_transactions` (int): Dining dollars transactions
- `bronco_bucks_transactions` (int): Bronco bucks transactions
- `credit_debit_transactions` (int): External payment transactions

Meal Period Distribution (4 columns)

- `breakfast_transactions` (int): Morning meal period
 - `lunch_transactions` (int): Midday meal period (typically highest)
 - `dinner_transactions` (int): Evening meal period
 - `late_night_transactions` (int): Late evening service
-

Financial Metrics (3 columns)

`estimated_daily_revenue` (float)

Estimated total daily revenue in dollars.

- **Calculation:** $\text{total_transactions} \times \text{avg_transaction_value}$
- **Usage:** Financial performance tracking

`avg_transaction_value` (float)

Average dollar value per transaction.

- **Default:** \$13.25 (all-you-care-to-eat effective value)
- **Usage:** Revenue per transaction analysis

`labor_cost_actual` (float)

Actual labor cost for predicted staffing.

- **Calculation:** $\text{total_actual_hours} \times \text{average_hourly_rate}$
 - **Usage:** Labor cost management
-

Operational Metrics (4 columns)

`transactions_per_meal_plan_holder` (float)

Daily transactions divided by meal plan holders.

- **Range:** 0.2 to 2.5
- **Usage:** Meal plan utilization efficiency

facility_capacity_utilization (float)

Percentage of facility capacity used.

- **Range:** 0.0 to 1.0
- **Calculation:** $\text{transactions} \div (\text{capacity} \times \text{daily_turns})$
- **Usage:** Capacity planning and bottleneck identification

peak_meal_period_volume (int)

Highest transaction count among meal periods.

- **Usage:** Peak capacity planning
-

Staffing Requirements (7 columns)

Role-Specific Staffing (6 columns)

- **actual_foh_general** (float): Front of house general staff hours
- **actual_foh_cashier** (float): Cashier staff hours
- **actual_kitchen_prep** (float): Kitchen prep staff hours
- **actual_kitchen_line** (float): Kitchen line staff hours
- **actual_dish_room** (float): Dish room staff hours
- **actual_management** (float): Management/supervision hours

total_actual_hours (float)

Sum of all role-specific staffing hours.

- **Range:** 8 to 80+ hours
 - **Usage:** Total labor requirement and primary ML target
-

Performance Metrics (3 columns)

revenue_per_labor_hour (float)

Revenue efficiency metric.

- **Calculation:** $\text{daily_revenue} \div \text{total_actual_hours}$
- **Usage:** Labor productivity analysis

transactions_per_labor_hour (float)

Transaction efficiency metric.

- **Calculation:** $\text{total_transactions} \div \text{total_actual_hours}$
- **Usage:** Operational efficiency tracking

labor_cost_percentage (float)

Labor cost as percentage of revenue.

- **Calculation:** $(\text{labor_cost} \div \text{daily_revenue}) \times 100$
 - **Range:** 15% to 45%
 - **Usage:** Cost control and profitability analysis
-

Parameter Impact Guide

High Impact Parameters (Primary Drivers)

total_enrollment_base & **yoy_growth_rate**

- **Impact:** Direct multiplier on all transaction volumes
- **Sensitivity:** 10% enrollment change = ~10% transaction change
- **Recommendation:** Use actual institutional data

residential_student_ratio

- **Impact:** Major driver of meal plan participation
- **Sensitivity:** 1% increase in residential ratio = ~3-5% more meal plan holders
- **Recommendation:** Monitor housing development plans

meal_plan_utilization_rates

- **Impact:** Directly affects actual vs potential transactions
- **Sensitivity:** 5% utilization change = 3-7% transaction change
- **Recommendation:** Survey actual student usage patterns

seasonal_multipliers & **weekly_patterns**

- **Impact:** Primary temporal variation drivers
- **Sensitivity:** 10% multiplier change = 8-12% volume change
- **Recommendation:** Validate against historical data

Medium Impact Parameters

weather_impacts & **event_impacts**

- **Impact:** Day-to-day variation overlay
- **Sensitivity:** Creates 5-25% daily variation around base levels
- **Recommendation:** Adjust based on local climate and event calendar

volume_scaling_factors (staffing)

- **Impact:** How staffing responds to volume changes
- **Sensitivity:** 0.1 change in scaling = 5-15% staffing change per volume increase
- **Recommendation:** Calibrate against actual labor scheduling

student_worker_availability

- **Impact:** Staffing adjustment multiplier
- **Sensitivity:** 10% availability change = 8-12% staffing hour change
- **Recommendation:** Based on academic calendar and student employment policies

Low Impact Parameters

payment_method_distribution

- **Impact:** Affects transaction categorization, not totals
- **Sensitivity:** Minimal impact on staffing predictions
- **Recommendation:** Use institutional payment system data

platform_popularity

- **Impact:** Affects platform-specific analysis, not total volume
- **Sensitivity:** No direct staffing impact
- **Recommendation:** Based on customer flow observations

Common Configuration Scenarios

Scenario 1: Standard Academic Year Analysis

python

```
config = {
    'start_date': '2024-08-15',
    'end_date': '2025-05-20',
    'student_population': {
        'total_enrollment_base': 31500, # Current CPP enrollment
        'yoy_growth_rate': 0.022,      # Modest growth
    }
}
```

Scenario 2: Growth Planning (Multi-Year)

python

```
config = {
    'start_date': '2024-01-01',
    'end_date': '2028-12-31',
    'student_population': {
        'total_enrollment_base': 31000,
        'yoy_growth_rate': 0.035,      # Aggressive growth
        'residential_student_ratio': 0.18, # Housing expansion
    }
}
```

Scenario 3: Crisis/Reduced Operations

python

```
config = {
    'academic_calendar': {
        'special_periods': {
            'fall_semester': {'multiplier': 0.75}, # Reduced capacity
            'spring_semester': {'multiplier': 0.70}, # Further reduction
        }
    },
    'staffing_model': {
        'scheduling_constraints': {
            'student_worker_availability': {
                'weekday_morning': 0.35, # Limited availability
                'finals_week': 0.25,     # Very limited
            }
        }
    }
}
```

Scenario 4: New Meal Plan Structure

python

```
config = {
    'meal_plans': {
        'plan_types': {
            'Unlimited_Plus': {
                'cost_per_semester': 3200,
                'typical_daily_usage': 2.8,
                'utilization_rate': 0.89,
                'student_distribution': 0.15,
            },
            # Modify existing plans...
        }
    }
}
```

Scenario 5: Facility Expansion Impact

python

```
config = {
    'facility_specs': {
        'maximum_simultaneous_capacity': 900, # Increased capacity
        'number_of_dining_platforms': 10,    # New platforms
    },
    'transaction_patterns': {
        'platform_popularity': {
            'New_Platform_1': 0.08,
            'New_Platform_2': 0.06,
            # Adjust existing platforms proportionally...
        }
    }
}
```

Troubleshooting & Validation

Common Issues

Low Transaction Volumes

Symptoms: Average daily transactions < 1,000 **Causes:**

- Low `total_enrollment_base`
- Low `residential_student_ratio`
- Low `meal_plan_utilization_rates`
- High proportion of break periods

Solutions:

- Verify enrollment parameters match institutional data
- Check seasonal multipliers for unrealistic values
- Validate meal plan participation rates

Extreme Staffing Hours

Symptoms: Daily staffing > 100 hours or < 20 hours regularly **Causes:**

- Incorrect `volume_scaling_factors`
- Unrealistic `minimum_coverage_hours`
- Wrong `student_worker_availability` settings

Solutions:

- Calibrate scaling factors against actual scheduling data
- Adjust minimum coverage based on operational requirements
- Validate availability constraints with HR policies

Revenue/Cost Imbalances

Symptoms: Labor cost > 60% of revenue or < 15% **Causes:**

- Wrong `average_hourly_rate`
- Incorrect `avg_transaction_value`
- Misaligned staffing parameters

Solutions:

- Update labor costs with current wage data
- Verify transaction values against actual revenue
- Benchmark staffing levels against industry standards

Validation Checklist

Data Quality Checks

- ☐ No negative transaction counts
- ☐ Staffing hours within reasonable bounds (5-80 hours/day)
- ☐ Payment methods sum to total transactions
- ☐ Meal periods sum to total transactions
- ☐ Labor cost percentage within industry norms (20-40%)

Operational Realism

- ☐ Higher transactions on weekdays vs weekends
- ☐ Peak volumes during lunch periods
- ☐ Lower activity during break periods
- ☐ Seasonal patterns match academic calendar
- ☐ Weather impacts directionally correct

Statistical Properties

- ☐ Transaction volumes follow reasonable distribution
- ☐ Day-to-day variation realistic (CV 0.15-0.35)
- ☐ Growth trends consistent with parameters
- ☐ No systematic biases in residuals

Performance Optimization

Memory Usage

- Large date ranges (>3 years) may require 1GB+ RAM
- Consider generating data in chunks for very long periods
- Save intermediate results for iterative analysis

Generation Speed

- Typical speed: 1,000-5,000 days per minute
- Multi-year datasets (1,000+ days) take 2-10 minutes
- Progress indicators provided for long generations

Output Size

- Typical file size: ~50KB per 100 days
- Annual dataset: ~200KB
- Multi-year datasets: 500KB to 2MB

Best Practices

Parameter Tuning

1. **Start with defaults** and modify incrementally
2. **Validate against actual data** when available
3. **Document all changes** and rationale
4. **Test edge cases** (holidays, emergencies, construction)
5. **Benchmark results** against industry standards

Model Training Preparation

1. **Generate multiple scenarios** for robust training
2. **Include seasonal cycles** (minimum 1 full academic year)
3. **Balance growth and stability** periods
4. **Validate feature distributions** before training
5. **Reserve holdout sets** for final model evaluation

Production Deployment

1. **Monitor prediction accuracy** against actual outcomes
2. **Update parameters** based on operational changes
3. **Retrain models** quarterly or after major changes
4. **Maintain configuration version control**
5. **Document parameter changes** with business justification

This manual provides comprehensive guidance for effectively using and customizing the Centerpointe Dataset Generator for various analytical and operational needs.