Practice: Evaluation Research

Session 01 - **ANOVA and ANCOVA**

psy112 - Evaluation Research

Faculty VI / UOL

Summer term 2025

Content

One-way ANOVA between person

Purpose: Compare means of one continuous DV across ≥3 independent groups (levels of one factor).

Hypotheses:

- H_0: All group means are equal ($\mu_1 = \mu_2 = ... = \mu_k$).
- H_a: At least one group mean differs.

Assumptions:

- Independence of observations.
- Normally distributed residuals (or DV within groups).
- Homogeneity of variances (homoscedasticity).

Example: Compare test scores (DV) of students who used different study methods (factor), where each student uses only one method.

One-way ANOVA within person - repeated measures, rmANOVA)

Purpose: Compare means of one continuous DV across $\geq$3 related measurements (conditions/time points) from the *same* observational units (individuals).

Hypotheses:

- H_0: All condition/time point means are equal ($\mu_1 = \mu_2 = ... = \mu_k$).
- H_a: At least one mean differs.

Assumptions:

- Independence of the observational units.
- Normally distributed DV at each measurement level.
- **Sphericity**: Variance of the differences between measurement pairs are equal. (If violated, use corrections e.g., Greenhouse-Geisser).

Example: Measuring the same participants' reaction times (DV) under three different distraction conditions (factor).

Two-way ANOVA between person

Purpose: Examine effects of *two* factors (A and B) on one continuous DV.

Effects tested in the model:

- Main Effect A: Does DV mean differ across levels of Factor A?
- Main Effect B: Does DV mean differ across levels of Factor B?
- Interaction Effect (A x B): Does the effect of Factor A depend on the level of Factor B (and vice versa)?

Assumptions: Independence, normality, nomogeneity of variances within each cell (combination of factor levels).

Example: How crop yield (DV) is affected by Fertilizer Type (Factor A) *and* Watering Frequency (Factor B).

ANCOVA (Analysis of Covariance)

Purpose: Compare group means (defined by factor(s)) on a DV after *statistically controlling* for the effect of one or more continuous covariates.

Key Idea: Adjusts group means to what they would be if all groups were equal on the covariate(s). Reduces error variance, potentially increasing power to detect effects.

Hypotheses: Tests equality of *adjusted* group means. Also tests significance of the covariate(s).

Additional Assumptions:

- Linear relationship between covariate(s) and DV.
- Homogeneity of regression slopes (covariate effect is the same in all groups - no covariate*factor interaction).

ANCOVA (Analysis of Covariance)

Example: Comparing post-therapy anxiety scores (DV) between treatment groups (factor), controlling for pre-therapy anxiety scores (covariate).

Python Package: `pingouin`

User-friendly stats package for common analyses in Python. Good for ANOVA/ANCOVA.

Key ANOVA/ANCOVA Functions:

- `pg.anova()`: Between-person ANOVA (1-way, N-way).
- `pg.rm_anova()`: Within-person / Repeated Measures ANOVA.
- `pg.mixed_anova()`: Mixed design ANOVA.
- `pg.ancova()`: ANCOVA.

Includes assumption tests and effect sizes. Works well with Pandas DataFrames.

Documentation: `pingouin-stats.org`

Python Package: pingouin

Example:

```
import pingouin as pg
# aov = pg.anova(dv='Score', between='Group', data=df)
# rm_aov = pg.rm_anova(dv='Score', within='Time', subject='Subject
    ', data=df)
ancv = pg.ancova(dv='Score', between='Group', covar='Covariate',
    data=df)
print(ancv)
```

Python: `statsmodels.stats.anova.anova_lm`

Computes ANOVA tables from fitted `statsmodels` linear models (e.g., from `ols`).

Features:

- Handles regression, ANOVA, ANCOVA models.
- Can compare nested models.
- Calculates Type I, II, or III Sum of Squares (important for interactions/unbalanced designs).

Use Case: Get F-tests and p-values for overall factors in a linear model after fitting it.

Documentation: `statsmodels.org`

Python: statsmodels.stats.anova.anova_lm

Example:

```
import statsmodels.api as sm
import statsmodels.formula.api as smf
# Assume 'model' is a fitted OLS model object:
# model = smf.ols('DV ~ C(FactorA) * C(FactorB)', data=df).fit()
anova_table = sm.stats.anova_lm(model, typ=2) # Type II SS
print(anova_table)
```

Python: `statsmodels.formula.api.ols`

Fits Ordinary Least Squares (OLS) linear models using R-style formulas.

Features:

- Formula Syntax (via `patsy`): e.g., `'DV   C(Group) + Covariate'`
- Handles categorical (`C()`) and continuous variables, interactions (`:`, `*`).
- Returns a fitted model object with coefficients, stats, residuals etc.

Use Case: The standard way to specify and fit models for ANOVA/ANCOVA before passing to `anova_lm`.

Documentation: `statsmodels.org`

Python: statsmodels.formula.api.ols

Example:

```
import statsmodels.formula.api as smf
# Define formula for ANCOVA
formula = 'Score ~ C(TreatmentGroup) + PreScore'
# Fit the model
model = smf.ols(formula, data=df).fit()
# Get summary (coefficients etc.)
print(model.summary())
```

Python: `patsy.ContrastMatrix`

Part of the `patsy` library (used by `statsmodels` formulas). Defines custom contrasts for categorical variables.

Purpose:

- Specify how categorical variables are coded numerically in the model matrix.
- Allows testing specific hypotheses (e.g., planned comparisons).
- Default is usually Treatment (dummy) coding. Other options include Sum, Helmert, Polynomial, or fully custom contrasts (you did not learn about these in psy111 - good to know that they exist).
- Sometimes needed for correct interpretation of Type III SS with interactions.

Use Case: Apply specific comparison structures within the `statsmodels` formula, e.g., `'DV   C(Group, Sum)'`.

Documentation: `patsy.readthedocs.io`

Python: `statsmodels.stats.multicomp.pairwise_tukeyhsd`

Performs Tukey's Honestly Significant Difference (HSD) post-hoc test.

Purpose:

- Used *after* a significant ANOVA F-test.
- Identifies which specific pairs of group means are significantly different.
- Controls the family-wise error rate (FWER) across all pairwise comparisons.

Input: Typically requires the data vector (DV) and the group labels vector.

Output: Table showing mean differences, confidence intervals, and adjusted p-values for each pair.

Documentation: `statsmodels.org`

Python: statsmodels.stats.multicomp.pairwise_tukeyhsd

Example:

```
from statsmodels.stats.multicomp import pairwise_tukeyhsd
# Assuming 'Score' is DV column, 'Group' is factor column in df
tukey_result = pairwise_tukeyhsd(endog=df['Score'], groups=df['
    Group'], alpha=0.05)
print(tukey_result)
```

Python Book for psy112

The companion Python book for the practical sessions is available online:

Resource Link

`psy112 Evaluation Research - Python Book`

Next session

Topic: Multilevel and SEM change score models

daniel.kristanto@uni-oldenburg.de & andrea.hildebrandt@uni-oldenburg.de