

Full Explanation of Line × Tester Analysis in R Studio

Line × Tester analysis is a mating design commonly used in plant breeding to evaluate:

- **General Combining Ability (GCA):** The average performance of a line or tester across all crosses.
- **Specific Combining Ability (SCA):** Performance of a specific line × tester combination.
- **Genetic Contributions:** Partitioning genetic variance into additive and non-additive components.

This method helps breeders identify superior parental lines and cross combinations for further improvement.

1. Data Structure and Preparation

Arrange your data in a dataframe with these columns:

- **replication:** Block or replication number
- **line:** Identifier for each line (female parent)
- **tester:** Identifier for each tester (male parent)
- **trait:** Measured trait (e.g., yield)

Example:

replication	line	tester	yield
1	1	1	60
1	1	2	56
1	2	1	63
...

2. Key Statistical Components

- **ANOVA Table:** Splits total variance into parts: lines, testers, line × tester (L×T) interaction, and error.
- **GCA Effects:** Calculated for each parent. High positive GCA = better average performance.
- **SCA Effects:** Deviation of a specific cross above/below GCA predictions.
- **Genetic Parameters:** Additive (GCA) and non-additive (SCA) variance components.

3. Recommended R Packages

- **agricolae**: Function `lineXtester` simplifies the process.
- **gpbStat**: Advanced, handles multiple traits—function `ltc` or `ltcmt`.
- **plantbreeding**: Function `line.test` for visualization and analysis.

4. Step-by-Step Analysis Using agricolae and gpbStat

A. Using agricolae

```
# Install and load
install.packages("agricolae")
library(agricolae)

# Suppose your data is in a dataframe 'ltdata'
# Columns: replication, line, tester, yield

# Line × Tester Analysis
result <- with(ltdata, lineXtester(replication, line, tester, yield))

# Results
print(result)
```

Output:

- **Means** for each cross.
- **ANOVA Table:**

Source	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Replication		
Lines
Testers
L×T
Error		

- **GCA and SCA Values** for each parent/cross.

B. Using gpbStat

```
# Install and load
install.packages("gpbStat")
library(gpbStat)

# Data: rcddlrc, columns: replication, line, tester, yield

# Analysis
result <- ltc(rcddlrc, replication, line, tester, yield)
```

```
# View results
result
```

Output Includes:

- Means by cross
- **Overall ANOVA**
- **Coefficient of Variation**
- **Genetic Variance (additive, dominance, etc.)**
- **GCA/SCA for lines and testers**
- **Proportional contributions**
- **Standard errors and critical differences**

5. Interpretation of Key Table Entries

a. ANOVA Table

- Large and significant ($p < 0.05$) L×T interaction: Certain crosses outperform beyond what GCA predicts.
- Significant GCA: Additive genetic effects are important.
- Significant SCA: Non-additive (dominant/epistatic) effects are significant.

b. GCA Effects Table

- Shows which lines and testers are generally good combiners.
- The higher (positive) the GCA, the better the parent for hybridization.

c. SCA Effects Table

- Shows specific cross combinations that perform exceptionally (positive SCA).

6. Worked Example

Suppose we have 3 lines and 2 testers, 2 replications, and grain yield as trait:

```
# Dummy data for demonstration
replication <- as.factor(rep(1:2, each=6))
line <- as.factor(rep(1:3, times=4))
tester <- as.factor(rep(rep(1:2, each=3), times=2))
yield <- c(60,62,58,64,61,65,59,63,60,67,62,64)

ltdata <- data.frame(replication, line, tester, yield)

# Analysis
library(agricolae)
result <- with(ltdata, lineXtester(replication, line, tester, yield))

# Extract and show tables
```

```
result$"ANOVA"
result$"GCA"
result$"SCA"
```

7. Advanced: Multiple Traits

For multiple traits, use `gpbStat`'s `ltcmt()` function.

Summary Table: Typical ANOVA Output Structure

Source	df	SS	MS	F	p-value
Replication	r-1				
Lines (L)	l-1				
Testers (T)	t-1				
L × T	(l-1)(t-1)				
Error	...				

- **GCA and SCA** tables follow.

Key References:

- agricolae help: `?lineXtester`
- [gpbStat Line × Tester vignette](#)
- Data and script examples: [Line-Tester-Analysis](#)

In summary:

- Arrange your data
- Use `lineXtester()` or `ltc()`
- Interpret output: ANOVA for significance, GCA for parent selection, SCA for best crosses
- Use post-hoc results (SCA, GCA) to select promising parents and hybrids for breeding programs

If you want a script for your own data or have questions about specific outputs, let me know!

