

A retrospective analysis of historical data from multi-environment trial evaluation for dry bean (*Phaseolus vulgaris* L.) in Michigan

Leonardo Volpato, Francisco E. Gomez, Evan Wright, and Scott Bales

Last compiled on 22 May, 2023

Contents

1	Appendix A - Supplemental Tables	6
2	Appendix B - Supplemental Figures	19
3	Appendix C - R codes	30
	Getting started	30
3.1	Packages	30
3.2	Data preparation	30
3.3	Descriptive Stats - Raw data	31
3.3.1	Box plot dist.	31
3.3.1.1	BB	35
3.3.1.2	NB	37
3.3.1.3	SR	39
3.3.2	GEI Comparisons (loc:year) - BB	40
3.3.3	GEI Comparisons (loc:year) - NB	41
3.3.4	GEI Comparisons (loc:year) - SR	42
3.3.5	Scatter plot	43
3.3.6	Outliers	44
3.3.7	Map of locations	45
3.3.8	Missing values	45
3.3.9	Coincidence Genotypes	46
3.4	Predicted by year mixed model analysis	47
3.5	Data preparation	47
3.5.1	By year mixed model analysis	47
3.5.1.1	Preparing dataset	49
3.5.2	Descriptive variance	49

3.5.2.1	Beans data set by year	49
3.5.2.2	Box plot distribution	50
3.6	Two Stage mixed model analysis	52
3.6.1	1st Stage mixed model analysis	52
3.6.1.1	Preparing dataset of Stage I for Stage II	54
3.6.2	2nd Stage mixed model analysis	55
3.6.3	By market classes	55
3.6.4	Descriptive variance	58
3.6.4.1	Box plot distribution	58
3.6.4.2	Phenotypic variance components	60
3.6.4.3	Genotyping performance	61
3.7	Descriptive MET	61
3.8	Analyzing MET - GGE	64
3.8.1	GGE biplot model	64
3.9	WAASB index	65
3.10	WAASBY index	65
4	Appendix D - R codes	65
4.1	MET analysis - Black beans	65
4.1.1	MET analysis - ASReml	65
4.1.2	MET analysis - lme4	66
4.1.3	Printing the model outputs	67
4.1.3.1	Likelihood Ratio Tests	67
4.1.3.2	Detailed parameters	67
4.1.3.3	Random effects	68
4.1.3.4	Variance components and genetic parameters	69
4.1.4	MET - GGE biplot	69
4.1.4.1	GGE ENV biplot	69
4.1.4.2	Biplot type 3: Which-won-where	71
4.1.5	Mean performance and stability analysis	72
4.1.5.1	Selection differentials	76
4.1.5.2	Scenarios of waasby estimation	80
4.1.5.3	Coincidence index of genotype selection	81
4.2	MET analysis - Navy beans	83
4.2.1	MET analysis - ASReml	83
4.2.2	MET analysis - lme4	84
4.2.3	Printing the model outputs	85

4.2.3.1	Likelihood Ratio Tests	85
4.2.3.2	Detailed parameters	85
4.2.3.3	Random effects	86
4.2.3.4	Variance components and genetic parameters	86
4.2.4	MET - GGE biplot	87
4.2.4.1	GGE ENV biplot	87
4.2.4.2	Biplot type 3: Which-won-where	89
4.2.5	Mean performance and stability analysis	90
4.2.5.1	Selection differentials	93
4.2.5.2	Scenarios of waasby estimation	97
4.2.5.3	Coincidence index of genotype selection	98
4.3	MET analysis - Red beans	100
4.3.1	MET analysis - ASReml	100
4.3.2	MET analysis - lme4	101
4.3.3	Printing the model outputs	102
4.3.3.1	Likelihood Ratio Tests	102
4.3.3.2	Detailed parameters	102
4.3.3.3	Random effects	103
4.3.3.4	Variance components and genetic parameters	103
4.3.4	MET - GGE biplot	104
4.3.4.1	GGE ENV biplot	104
4.3.4.2	Biplot type 3: Which-won-where	106
4.3.5	Mean performance and stability analysis	107
4.3.5.1	Selection differentials	111
4.3.5.2	Scenarios of waasby estimation	114
4.3.5.3	Coincidence index of genotype selection	115
5	Appendix E - R codes	117
5.1	Multi-trait Multi-env analysis	117
5.2	Descriptive Stats - Multi-trait data	117
5.2.1	Data set	117
5.2.2	Box plot distribution	118
5.2.3	Scatter plot distribution GY	121
5.2.4	Scatter plot distribution DM	122
5.2.5	Scatter plot distribution PH	123
5.2.6	Scatter plot distribution LD	124
5.2.7	Missing values	124

5.2.8	Linear phenotypic correlation	125
5.3	Multi-trait stability index and selection gains	126
5.4	Black beans	126
5.4.1	Multi-trait index selection	127
5.4.2	Model outputs	127
5.4.2.1	Variance plot	127
5.4.2.2	Likelihood-ratio test	128
5.4.2.3	Variance components	129
5.4.2.4	Genetic parameters	129
5.4.2.5	Models details	130
5.4.2.6	Fixed effects	130
5.4.2.7	Enviroment means	131
5.4.3	Selection Description	131
5.4.3.1	Genotype ranking	131
5.4.3.2	Mean performance and stability for GY	133
5.4.3.3	Mean performance and stability for DM	134
5.4.3.4	Mean performance and stability for PH	135
5.4.3.5	Mean performance and stability for LD	136
5.5	Navy beans	136
5.5.1	Multi-trait index selection	136
5.5.2	Model outputs	137
5.5.2.1	Variance plot	137
5.5.2.2	Likelihood-ratio test	138
5.5.2.3	Variance components	139
5.5.2.4	Genetic parameters	139
5.5.2.5	Models details	140
5.5.2.6	Fixed effects	140
5.5.2.7	Enviroment means	141
5.5.3	Selection Description	141
5.5.3.1	Genotype ranking	141
5.5.3.2	Mean performance and stability for GY	143
5.5.3.3	Mean performance and stability for DM	144
5.5.3.4	Mean performance and stability for PH	145
5.5.3.5	Mean performance and stability for LD	146
5.6	Red beans	146
5.6.1	Multi-trait index selection	146

5.6.2	Model outputs	147
5.6.2.1	Variance plot	147
5.6.2.2	Likelihood-ratio test	148
5.6.2.3	Variance components	149
5.6.2.4	Genetic parameters	149
5.6.2.5	Models details	150
5.6.2.6	Fixed effects	150
5.6.2.7	Enviroment means	151
5.6.3	Selection Description	151
5.6.3.1	Genotype ranking	151
5.6.3.2	Mean performance and stability for GY	153
5.6.3.3	Mean performance and stability for DM	154
5.6.3.4	Mean performance and stability for PH	155
5.6.3.5	Mean performance and stability for LD	156
5.7	MTME mixed model	156
5.7.1	Black beans	156
5.7.2	Navy beans	158
5.7.3	Red beans	160
5.8	MTME Var Comp and corr	161
6	Package References	166
	References	168

This is the supplemental material for the manuscript **A retrospective analysis of historical data from multi-environment trial evaluation for dry bean (*Phaseolus vulgaris* L.) in Michigan.**

The Supplemental material is divided into six main sections:

1. The first section includes the Supplemental Tables.
2. The second section includes the Supplemental Figures.
3. The third section provides the R code needed to reproduce the descriptive analysis and methods presented in this manuscript.
4. The fourth section offers the R code to replicate the Multi-Environment Trials (MET) analysis.
5. The fifth section contains the R code for reproducing the Multi-Trait Multi-Environment analysis.
6. The sixth and final section lists the references and literature cited throughout this manuscript.

1 Appendix A - Supplemental Tables

Supplemental Table S1: Commercial names (ComName), code, market class (MktCl), source, release year (RelYear), yield (RangeYield) and maturity (RangeMat) range observed in the Michigan Dry Bean Performance Trials analyzed in the study.

ComName	Code	MktCl	Source	RelYear	RangeYield	RangeMat
213SP	N1	NB	Gentec	NA	1327.08-3808.82	104
ACE	B1	BB	ADM	2019	1602.9-4260.38	85 - 90
ADAMS	B2	BB	MSU	2020	1483.23-5718.08	87 - 102
ALPENA	N2	NB	MSU	2014	1598.2-3581.55	NA
APEX	N3	NB	AAFC-HARROW	2011	1383.51-4483.24	87 - 104
ARGOSY	N4	NB	ACUG	2016	1171.66-4475.56	87 - 102
ARMADA	N5	NB	ProVita Seeds	2019	1286.15-4927.81	86 - 100
PVT13505	B3	BB	ProVita Seeds	NA	1355.15-4726.52	NA
BL13SR1-1	B4	BB	Gentec	NA	1695.75-4286.68	98 - 103
BL1402-15	B5	BB	USDA-ARS	NA	1708.5-5097.63	95 - 101
BL15430	B6	BB	MSU	NA	1912.4-4159.36	NA
BL15442	B7	BB	MSU	NA	2185.21-4289.16	NA
BL15447	B8	BB	MSU	NA	1593.92-4817.12	NA
BL16501	B9	BB	MSU	NA	2184.43-4459.89	NA
BL16504	B10	BB	MSU	NA	1704.39-5206.16	NA
BL16506	B11	BB	MSU	NA	1385.56-4254.7	NA
BL16507	B12	BB	MSU	NA	1937.44-4114.39	NA
BL17220	B13	BB	MSU	NA	1716.08-5010.39	96 - 102
BL1726-2	B14	BB	USDA-ARS	NA	1327.08-4677.09	85 - 92
BL17536	B15	BB	MSU	NA	2024.01-4909.58	NA
BL17691	B16	BB	MSU	NA	2195.96-4982.54	NA
BL17922	B17	BB	MSU	NA	1595.69-5468.59	99 - 101
BL18094173	B18	BB	ADM	NA	2333.06-4501.36	101
BL18201	B19	BB	MSU	NA	1891.45-5501.6	95 - 101
BL18204	B20	BB	MSU	NA	1424.79-5611.63	90 - 98
BL19309	B21	BB	MSU	NA	1507.6-4657.46	87 - 95
BL19330	B22	BB	MSU	NA	1952.66-4736.5	84 - 98
BL19344	B23	BB	MSU	NA	1893.22-5406.24	86 - 104
BL20536	B24	BB	MSU	NA	2486.05-3905.59	103
BL20547	B25	BB	MSU	NA	1355.45-4561.57	87 - 93
BL20591	B26	BB	MSU	NA	1586.82-4802.97	86 - 93
BL20597	B27	BB	MSU	NA	2380.71-4160.69	84 - 92
BL20599	B28	BB	MSU	NA	2616.34-4423.61	102
BL21708	B29	BB	MSU	NA	1771.54-4972.09	100
BL21710	B30	BB	MSU	NA	2590.07-4589.62	101
BL21714	B31	BB	MSU	NA	2217.06-3910.32	100
BL3033350	B32	BB	ADM	NA	1708.92-3438.01	97
BL3035411	B33	BB	ADM	NA	2912.65-3975.99	98
BL3036368	B34	BB	ADM	NA	1174.63-5310.16	93 - 100
BL3036381	B35	BB	ADM	NA	2068.64-5125.49	87 - 105
BL5058320	B36	BB	ADM	NA	1700.93-3758.7	98
BL7071259	B37	BB	ADM	NA	2402.93-4766.15	96
BL7072269	B38	BB	ADM	NA	2030.03-4444.62	98
BLACKBEAR	B39	BB	ProVita Seeds	2018	974.19-4937.31	90 - 105
BLACKBEARD	B40	BB	ProVita Seeds	2019	1156.46-5436.5	90 - 101
BLACKCAT	B41	BB	ADM	2014	1517.23-4965.22	NA
BLACKTAILS	B42	BB	ProVita Seeds	2017	889.94-5179.46	87 - 102
BLIZZARD	N6	NB	ProVita Seeds	2017	1065.87-4563.69	88 - 100
BOUNTY	N7	NB	ProVita Seeds	2019	1009.56-4655.82	88 - 100
CALDERA	R1	SR	ProVita Seeds	2018?	1196.06-5385.04	97 - 101
CAYENNE	R2	SR	MSU	2017	1545.11-4682.2	84 - 100
CORAL	R3	SR	MSU	NA	854.25-4077.7	102
DS105WO	N8	NB	DOW	NA	1946.32-4091.65	NA
ECLIPSE	B43	BB	NDSU	2004	1277.47-4595.38	85 - 97
EX1701	N9	NB	TVS	NA	905.46-4163.67	NA

EX1702	N10	NB	TVS	NA	1153.41-4098.98	95 - 101
EX1703	N11	NB	TVS	NA	1656.7-4457.85	NA
EX1708	N12	NB	TVS	NA	670.86-4596.67	98 - 101
EX1711	N13	NB	TVS	NA	1020.8-5158.16	94 - 98
EX1801	N14	NB	TVS	NA	1760.95-4096.31	93 - 98
EX1802	N15	NB	TVS	NA	991.05-4569.08	85 - 96
EX1803	N16	NB	TVS	NA	1191.12-4448.04	85 - 102
EX1804	N17	NB	TVS	NA	1104.54-4184.9	85 - 98
EX1914	N18	NB	TVS	NA	1143.87-4970.85	95 - 100
EX2109	N19	NB	TVS	NA	1118.41-3627.57	97
GTSOB-1723-03	N20	NB	Gentec	NA	1748.11-3260.29	NA
INDI	N21	NB	ADM	2010	1131.44-4030.38	90 - 95
LIBERTY	N22	NB	ProVita Seeds	2021	1191.07-5322.3	87 - 100
LORETO	B44	BB	ProVita Seeds	2007	1007.97-4569.47	NA
MEDALIST	N23	NB	ProVita Seeds	2007	1478.68-4978.47	85 - 101
MERLIN	N24	NB	ProVita Seeds	2010	998.54-4561.73	90 - 102
MERLOT	R4	SR	USDA/MSU	2002	1356.14-4447.94	NA
MIST	N25	NB	Guelph	2013	921.13-4301	NA
NA14229	N26	NB	MSU	NA	1334.35-4316.25	NA
NA15341	N27	NB	MSU	NA	2200.33-4256.5	NA
NA16401	N28	NB	MSU	NA	1270.06-3186.23	NA
NA16405	N29	NB	MSU	NA	1923-3834.66	NA
NA17504	N30	NB	MSU	NA	1864.97-4145.36	NA
NA17506	N31	NB	MSU	NA	1272.67-4366.3	NA
NA18102	N32	NB	MSU	NA	1096.13-3630.12	NA
NA18103	N33	NB	MSU	NA	1228.86-4517.17	86 - 95
NA18109	N34	NB	MSU	NA	1293.85-4939.4	NA
NA19226	N35	NB	MSU	NA	2086.6-4376.18	86 - 93
NA19246	N36	NB	MSU	NA	1447.5-4435.73	86 - 95
NA19253	N37	NB	MSU	NA	1520.92-4810.69	87 - 99
NA19285	N38	NB	MSU	NA	1241.64-4030.95	84 - 101
NA20388	N39	NB	MSU	NA	1682.97-4325.13	87 - 92
NA20395	N40	NB	MSU	NA	1241.55-4281.97	98
NA20404	N41	NB	MSU	NA	1536.18-4060.15	85 - 90
NA21511	N42	NB	MSU	NA	1147.41-3920.51	101
NA21525	N43	NB	MSU	NA	1152.87-4009.62	100
NAUTICA	N44	NB	AAFC-HARROW	2004	1020.64-4230.28	86 - 102
NDPOLAR	N45	NB	NDSU	2022	1465.99-3674.75	102
NDTWILIGHT	B45	BB	NDSU	2020	1270.97-4140.56	84 - 95
NIMBUS	B46	BB	ProVita Seeds	2021	1882.19-5708.71	89 - 103
OACVortex	B47	BB	NA	2019	1539.75-4261.79	101
PVT06063	N46	NB	ProVita Seeds	NA	1848.4-3805.14	NA
PVT12039	N47	NB	ProVita Seeds	NA	904.55-5349.49	87 - 102
PVT12062	N48	NB	ProVita Seeds	NA	1352.32-5853.77	NA
PVT12063	N49	NB	ProVita Seeds	NA	1436.99-5235.81	NA
PVT12064	N50	NB	ProVita Seeds	NA	554.77-4168.64	NA
PVT13049	N51	NB	ProVita Seeds	NA	1739.78-3849.08	NA
PVT13058	N52	NB	ProVita Seeds	NA	988.66-4534.09	NA
PVT13066	N53	NB	ProVita Seeds	NA	1638.17-4497.35	NA
PVT13490	B48	BB	ProVita Seeds	NA	2128.87-4264.54	NA
PVT13496	B49	BB	ProVita Seeds	NA	1526.42-3442.4	NA
PVT13503	B50	BB	ProVita Seeds	NA	1104.68-3457.17	NA
PVT14068	N54	NB	ProVita Seeds	NA	1273.85-5141.5	87 - 103
PVT14069	N55	NB	ProVita Seeds	NA	2043-4228.37	NA
PVT14075	N56	NB	ProVita Seeds	NA	1692.95-3804.3	104
PVT14078	N57	NB	ProVita Seeds	NA	1622.35-4479.66	87 - 100
PVT14080	N58	NB	ProVita Seeds	NA	1078.05-4217.83	96 - 103
PVT14081	N59	NB	ProVita Seeds	NA	1954.85-3687.03	NA
PVT14084	N60	NB	ProVita Seeds	NA	921.7-4564.38	87 - 98
PVT14089	N61	NB	ProVita Seeds	NA	1673.84-4117.56	96 - 100
PVT14496	B51	BB	ProVita Seeds	NA	2017.05-5255.23	NA
PVT14498	B52	BB	ProVita Seeds	NA	1869.43-4938.29	NA
PVT14504	B53	BB	ProVita Seeds	NA	2059.98-5023.2	NA
PVT14505	B54	BB	ProVita Seeds	NA	2881.32-4994.5	NA
PVT14510	B55	BB	ProVita Seeds	NA	1477.68-3779.63	NA

PVT14522	B56	BB	ProVita Seeds	NA	1692.75-4231.31	NA
PVT14531	B57	BB	ProVita Seeds	NA	2196.46-3956.91	0 - 95
PVT15610	B58	BB	ProVita Seeds	NA	1369-5461.59	85 - 104
PVT15619	B59	BB	ProVita Seeds	NA	1374.37-5601.63	87 - 105
PVT15629	B60	BB	ProVita Seeds	NA	2890.2-4122.78	87 - 94
PVT16113	N62	NB	ProVita Seeds	NA	1327.08-4098.72	100
PVT16590	B61	BB	ProVita Seeds	NA	2232.82-4940.66	89 - 103
PVT16598	B62	BB	ProVita Seeds	NA	2441.91-4340.28	101
PVT16648	B63	BB	ProVita Seeds	NA	2016.09-4804.76	85 - 104
PVT16686	R5	SR	ProVita Seeds	NA	1307.27-5053.8	85 - 98
PVT17704	B64	BB	ProVita Seeds	NA	1750.84-4763.96	97 - 104
PVT17708	B65	BB	ProVita Seeds	NA	1726.98-4357.16	94 - 101
PVT17715	B66	BB	ProVita Seeds	NA	1988-6014.39	87 - 101
PVT17724	B67	BB	ProVita Seeds	NA	1982.1-4651.46	92 - 98
PVT17751	B68	BB	ProVita Seeds	NA	2465.45-5501.17	85 - 100
PVT17822	R6	SR	ProVita Seeds	NA	2360.38-4868.28	100
PVT17835	R7	SR	ProVita Seeds	NA	654.06-3876.72	100 - 101
PVT17837	R8	SR	ProVita Seeds	NA	1152.23-4547.04	88 - 101
PVT17839	R9	SR	ProVita Seeds	NA	1116.84-4342.43	88 - 101
PVT17875	R10	SR	ProVita Seeds	NA	1826.18-4885.76	84 - 92
PVT19837	R11	SR	ProVita Seeds	NA	2580.93-4075.81	97
SR20627	R12	SR	MSU	NA	1566.02-4226.28	87 - 103
SR20667	R13	SR	MSU	NA	1641.67-4772.77	102
SR20669	R14	SR	MSU	NA	2638.4-4766.99	104
REXETER	N63	NB	Guelph	2011	1178.3-5089.48	NA
ROGUE	N64	NB	Guelph	2020?	1541.54-4112.92	0 - 92
ROSETTA	R15	SR	MSU	2012	1458.75-4738.55	86 - 94
RUBY	R16	SR	ProVita Seeds	2014	1088.14-5052.16	87 - 100
SHANIA	B69	BB	ADM	2008	1629.88-3656.14	NA
SHOCK	N65	NB	ACUG	2017	1548.47-4274.91	86 - 102
SPECTRE	B70	BB	ProVita Seeds	2019	1445.67-5859.47	90 - 102
SR16503	R17	SR	MSU	NA	1419.17-4401.98	NA
SR17603	R18	SR	MSU	NA	1375.77-5617.34	NA
SR17604	R19	SR	MSU	NA	1189.34-4722.51	85 - 103
SR17605	R20	SR	MSU	NA	1326.57-4372.72	NA
SV1893GH	N66	NB	Seminis Seeds	2015	1358.73-4918.82	87 - 104
T9905	N67	NB	TVS/Bred by Hyland	NA	1400.89-4338.23	NA
VALIANT	N68	NB	ProVita Seeds	2019	1045.13-4751.38	86 - 98
VICTORY	N69	NB	ProVita Seeds	2021	1238.93-4889.14	87 - 98
VIGILANT	N70	NB	ProVita Seeds	2012	922.36-4562.16	86 - 98
VIPER	R21	SR	ProVita Seeds	2015	1200.75-5624.94	19 - 102
VISTA	N71	NB	Gentec	1991	1752.27-3372.03	NA
ZENITH	B71	BB	MSU	2014	1172.34-4931.87	87 - 102
ZORRO	B72	BB	MSU	2008	1157.27-5047.25	88 - 102

Note:

NA = Missing values

Supplemental Table S2: Summary of the results for random- and fixed-effects and overall means for each locations of the grain yield (GY: 2017 - 2022; GY2: only 2021), days to maturity (DM), plant height (PH) and lodging (LD) traits evaluated in the study.

Market.Class	Parameters	GY	GY2	DM	PH	LD
BB	LRTG	12.0***	25.1***	30.2***	6.15*	12.3***
BB	LRTGE	216***	3.95*	9.7**	0.4ns	1.33ns
BB	E-MS	1373806***	1637595***	152***	23.6**	3.57***
BB	E/B-MS	1775719***	740206***	12.2***	16.9***	0.7**
BB	BA	3004.17	2835	89.9	16.9	1.32
BB	HU	3461.59	NA	NA	NA	NA
BB	SA	3321.17	3362	91	18.4	1.96
BB	TU	3233.21	3402	94.4	19.1	1.91
BB	Mean	3255.05+-19.84	3199.25+-30.37	91.77+-0.19	18.14+-0.15	1.73+-0.04
NB	LRTG	17.2***	17.9***	25.4***	12.1***	36***
NB	LRTGE	83.4***	7.79**	7.23**	6.15*	4.5*
NB	E-MS	1944731***	1493686***	120***	41.2***	2.37***
NB	E/B-MS	1735124***	320888**	12.7***	7.38*	0.92***
NB	BA	2654.03	2732	90.6	17	1.46
NB	HU	3094	NA	NA	NA	NA
NB	SA	2978.02	3094	89.6	18.5	2.2
NB	TU	2970.59	3195	93.6	19.6	1.97
NB	Mean	2924+-17.22	3009.36+-26.04	91.28+-0.16	18.37+-0.15	1.87+-0.04
SR	LRTG	3.33*	0	2.29ns	8.13**	11.3***
SR	LRTGE	46.3***	11.22***	2.58ns	3.83ns	0.76ns
SR	E-MS	1299967***	1722395***	87.2***	13*	1.24*
SR	E/B-MS	353243**	242390ns	7.61*	6.78ns	1.49***
SR	BA	2732.2	2584	89.2	17.5	1.37
SR	HU	3267.9	NA	NA	NA	NA
SR	SA	3223.4	3335	89.9	17.6	2.01
SR	TU	3367.3	3546	93.5	19.6	2.24
SR	Mean	3147.37+-32.29	3157.34+-53.6	90.86+-0.25	18.23+-0.24	1.87+-0.07

Note:

* Significant at $P < 0.05$. ** Significant at $P < 0.01$. *** Significant at $P < 0.001$. ns, nonsignificant.

† LRTg and LRTge, Likelihood ratio tests for genotype and interaction gxe. ‡ Vg, Vge and Vr, variance components for genotype, interaction, and residuals, respectively. § E-MS and E/B-MS, mean squares for environment and and block-within-environment, respectively. ¶ Mean, grand mean; # Standart error. NA = Missing values

Supplemental Table S3: Estimated variance components and genetic parameters for grain yield (GY), days to maturity (DM), plant height (PH) and lodging (LD) for 37 Black, 37 Navy and 14 Red Small beans genotypes evaluated in BA, SA, and TU environments during 2021 growing season only.

Black Beans

REML	GY2	DM	PH	LD
$\hat{\sigma}_g^2$	68572 (32%) [‡]	2.86 (39.5%)	0.63 (10.5%)	0.063 (17.6%)
$\hat{\sigma}_{ge}^2$	17115 (8%)	0.8 (11%)	0.18 (3.0%)	0.02 (5.32%)
$\hat{\sigma}_\varepsilon^2$	128582 (60%)	3.58 (49.5%)	5.17 (86.4%)	0.27 (77.1%)
$\hat{\sigma}_p^2$	214269	7.24	5.98	0.356
h_g^2	0.32	0.395	0.105	0.176
R_{gei}^2	0.08	0.11	0.0305	0.0532
h_{gm}^2	0.81	0.835	0.561	0.682
<i>Acc</i>	0.90	0.914	0.749	0.826
r_{ge}	0.12	0.182	0.0341	0.0645
<i>CVg</i>	8.19	1.84	4.37	14.5
<i>CVr</i>	11.21	2.06	12.5	30.3
<i>CVg/CVr</i>	0.73	0.894	0.349	0.478

Navy Beans

REML	GY2	DM	PH	LD
$\hat{\sigma}_g^2$	50195 (25.72%) [‡]	1.97 (32%)	1.36 (19.1%)	0.195 (39.3%)
$\hat{\sigma}_{ge}^2$	22779 (11.67%)	0.63 (10.2%)	0.81 (11.4%)	0.04 (7.19%)
$\hat{\sigma}_\varepsilon^2$	122206 (62.61%)	3.56 (57.8%)	4.95 (69.5%)	0.27 (53.5%)
$\hat{\sigma}_p^2$	195181	6.17	7.12	0.496
h_g^2	0.26	0.32	0.191	0.393
R_{gei}^2	0.12	0.102	0.114	0.0719
h_{gm}^2	0.74	0.796	0.666	0.851
<i>Acc</i>	0.86	0.892	0.816	0.923
r_{ge}	0.16	0.15	0.141	0.118
<i>CVg</i>	7.45	1.54	6.35	23.6
<i>CVr</i>	11.62	2.07	12.1	27.6
<i>CVg/CVr</i>	0.64	0.744	0.524	0.856

Red Beans

REML [§]	GY2	DM	PH	LD
$\hat{\sigma}_G^2$	0.1 (0.01%) [‡]	0.61 (12.2%)	1.81 (25.3%)	0.18 (34.6%)
$\hat{\sigma}_{ge}^2$	75950 (31.4%)	0.66 (13.2%)	0.35 (4.9%)	0.03 (5.0%)
$\hat{\sigma}_\varepsilon^2$	165822 (68.59%)	3.72 (74.6%)	4.99 (69.8%)	0.32 (60.4%)
$\hat{\sigma}_P^2$	241772	4.98	7.142	0.53
h_G^2	0.00	0.12	0.2531	0.35
R_{gei}^2	0.31	0.13	0.04876	0.05
h_{gm}^2	0.00	0.54	0.7728	0.84
Acc	0.00	0.73	0.8791	0.91
r_{ge}	0.31	0.15	0.06528	0.07
CVg	0.00	0.86	7.307	22.9
CVr	12.90	2.12	12.13	30.2
CVg/CVr	0.00	0.4	0.6022	0.76

$\hat{\sigma}_G^2$, genotypic variance; $\hat{\sigma}_{GE}^2$, variance of $G \times E$ interaction; $\hat{\sigma}_\varepsilon^2$ residual variance; $\hat{\sigma}_P^2$ phenotypic variance; h_G^2 broad-sense heritability; R_{gei}^2 coefficient of determination for the genotype-vs-environment interaction effects; h_{gm}^2 heritability of the genotypic mean; Acc accuracy of genotype selection; r_{GE} correlation between genotypic values across environments; CVg genotypic coefficient of variation; CVr residual coefficient of variation; CVg/CVr and is the ratio between genotypic and residual coefficient of variation.[‡] Parenthetical values indicate the percentage of the observed phenotypic variance $\hat{\sigma}_P^2$. The genetic parameters were estimated by $h_G^2 = \frac{\hat{\sigma}_G^2}{\hat{\sigma}_G^2 + \hat{\sigma}_{GE}^2 + \hat{\sigma}_\varepsilon^2}$; where $\hat{\sigma}_G^2$ is the genotypic variance; $\hat{\sigma}_{GE}^2$ is the genotype-by-environment interaction variance; and $\hat{\sigma}_\varepsilon^2$ is the residual variance. Coefficient of determination of the interaction effects: $R_{gei}^2 = \frac{\hat{\sigma}_{GE}^2}{\hat{\sigma}_G^2 + \hat{\sigma}_{GE}^2 + \hat{\sigma}_\varepsilon^2}$; heritability on the mean basis: $h_{gm}^2 = \frac{\hat{\sigma}_G^2}{[\hat{\sigma}_G^2 + \hat{\sigma}_{GE}^2/e + \hat{\sigma}_\varepsilon^2/(eb)]}$ where e and b are the number of environments and blocks, respectively; The accuracy of selection: $Acc = \sqrt{h_{GM}^2}$; genotype-environment correlation $r_{GE} = \frac{\hat{\sigma}_g^2}{\hat{\sigma}_G^2 + \hat{\sigma}_{GE}^2}$; genotypic coefficient of variation $CVg = \left(\sqrt{\hat{\sigma}_g^2}/\mu\right) \times 100$ residual coefficient of variation estimated: $CVr = \left(\sqrt{\hat{\sigma}_\varepsilon^2}/\mu\right) \times 100$ where μ is the grand mean; ratio between genotypic and residual coefficient of variation: CVg/CVr .

Supplemental Table S4: Selection differential of selected genotypes for mean performance and stability (WAASB index) at black beans. Selected genotypes: B14, B19, B21, B22, B25, B26, B51, B52, B53, B10, B17, B23, B61, B63, B66, B68, B2, B40, B46, B70, B24, B27, B30, B33, B60, B7.

Geno	Mean performance			Stability (WAASB)		
	BLUPs	Overhall	SD (%)	WAASB	Overhall	SD (%)
B10	3391	3255	4.1787	0.2971	5.152	-94.233
B14	3283	3255	0.8469	2.7163	5.152	-47.278
B17	3281	3255	0.8017	5.4405	5.152	5.598
B19	3316	3255	1.8723	3.6947	5.152	-28.288
B2	3401	3255	4.4717	3.1153	5.152	-39.534
B21	3458	3255	6.2395	3.3874	5.152	-34.253
B22	3285	3255	0.9310	3.1503	5.152	-38.854
B23	3433	3255	5.4621	2.9277	5.152	-43.174
B24	3336	3255	2.4965	2.9964	5.152	-41.841
B25	3398	3255	4.3767	4.8411	5.152	-6.038
B26	3448	3255	5.9262	1.8807	5.152	-63.497
B27	3346	3255	2.8048	2.6129	5.152	-49.285
B30	3373	3255	3.6317	5.7096	5.152	10.821
B33	3344	3255	2.7175	5.7243	5.152	11.106
B40	3377	3255	3.7361	1.5201	5.152	-70.496
B46	3526	3255	8.3099	3.2089	5.152	-37.717
B51	3397	3255	4.3524	1.7499	5.152	-66.035
B52	3359	3255	3.1927	3.1465	5.152	-38.927
B53	3382	3255	3.9096	2.8632	5.152	-44.428
B60	3338	3255	2.5341	2.7706	5.152	-46.224
B61	3381	3255	3.8636	2.4342	5.152	-52.753
B63	3362	3255	3.2767	2.9257	5.152	-43.214
B66	3348	3255	2.8490	1.0713	5.152	-79.207
B68	3523	3255	8.2229	2.7228	5.152	-47.152
B7	3282	3255	0.8249	3.7108	5.152	-27.975
B70	3372	3255	3.6026	1.2369	5.152	-75.993

Supplemental Table S 5: Selection differential of selected genotypes for mean performance and stability (WAASB index) at navy beans. Selected genotypes: N37, N39, N41, N58, N64, N10, N21, N33, N3, N57, N68, N54, N69, N70, N23, N4, N44, N47, N5, N6, N60, N7, N27, N29, N59.

Geno	Mean performance			Stability (WAASB)		
	BLUPs	Overhall	SD (%)	WAASB	Overhall	SD (%)
N10	2955	2924	1.0472	2.3226	3.912	-40.628
N21	2968	2924	1.5017	1.4534	3.912	-62.846
N23	3056	2924	4.5210	1.7097	3.912	-56.294
N27	3002	2924	2.6588	2.6325	3.912	-32.706
N29	2923	2924	-0.0436	1.4228	3.912	-63.629
N3	3083	2924	5.4533	2.7463	3.912	-29.794
N33	2964	2924	1.3811	2.9489	3.912	-24.618
N37	3049	2924	4.2847	4.6672	3.912	19.310
N39	2991	2924	2.3038	1.5480	3.912	-60.428
N4	3033	2924	3.7392	0.9353	3.912	-76.092
N41	3020	2924	3.2884	3.2316	3.912	-17.391
N44	2935	2924	0.3644	1.1779	3.912	-69.890
N47	3129	2924	7.0024	2.1505	3.912	-45.026
N5	3058	2924	4.5735	1.6651	3.912	-57.435
N54	3056	2924	4.5117	1.8380	3.912	-53.015
N57	3032	2924	3.6901	2.8651	3.912	-26.758
N58	2968	2924	1.4933	3.4484	3.912	-11.849
N59	2963	2924	1.3346	1.8183	3.912	-53.518
N6	2995	2924	2.4132	1.4008	3.912	-64.192
N60	3046	2924	4.1714	0.8606	3.912	-78.000
N64	3036	2924	3.8222	3.9567	3.912	1.146
N68	2990	2924	2.2719	3.6703	3.912	-6.176
N69	3141	2924	7.4060	1.7844	3.912	-54.386
N7	3130	2924	7.0594	1.2604	3.912	-67.779
N70	2928	2924	0.1372	2.8340	3.912	-27.555

Supplemental Table S6: Selection differential of selected genotypes for mean performance and stability (WAASB index) at red beans. Selected genotypes: R10, R13, R18, R1, R2, R21, R11.

Geno	Mean performance			Stability (WAASB)		
	BLUPs	Overhall	SD (%)	WAASB	Overhall	SD (%)
R1	3251	3147	3.2992	3.110	6.065	-48.72
R10	3180	3147	1.0359	5.367	6.065	-11.52
R11	3174	3147	0.8525	7.646	6.065	26.06
R13	3160	3147	0.3927	5.114	6.065	-15.68
R18	3278	3147	4.1476	3.705	6.065	-38.92
R2	3209	3147	1.9472	4.060	6.065	-33.06
R21	3313	3147	5.2734	3.143	6.065	-48.19

Supplemental Table S7: BLUP-based indexes for selecting genotypes with performance and stability at black beans.

GEN	Y	HMGV	HMGV_R	RPGV	RPGV_R	HMRPGV	HMRPGV_R	WAASB	WAASB_R	WAASBY	WAASBY_R
B1	3122	3049	54	3083	55	3068	54	2.67	17	67.58	35
B10	3300	3473	10	3489	10	3487	10	0.30	1	89.22	3
B11	3369	2884	62	2987	61	2896	62	10.27	66	41.97	69
B12	2985	3083	53	3121	54	3103	53	5.29	42	61.40	46
B13	3666	3118	52	3136	52	3127	52	2.08	12	70.88	30
B14	3305	3282	32	3296	35	3289	33	2.72	18	74.93	23
B15	3539	3310	30	3419	23	3314	31	12.17	70	52.80	60
B16	3644	3128	51	3228	40	3141	51	12.63	71	44.54	67
B17	2775	3232	35	3278	37	3254	35	5.44	43	67.12	37
B18	3537	3381	25	3431	21	3376	26	8.90	63	62.01	44
B19	3459	3313	29	3351	31	3328	28	3.69	35	74.45	24
B2	3507	3490	8	3509	8	3502	7	3.12	26	81.90	12
B20	3526	2981	58	3006	59	2998	58	3.28	30	62.96	43
B21	3296	3607	4	3619	4	3615	4	3.39	32	85.08	7
B22	3613	3281	33	3301	34	3289	34	3.15	28	73.89	26
B23	2969	3569	6	3581	6	3569	6	2.93	23	84.64	9
B24	2865	3392	23	3403	29	3391	23	3.00	25	77.82	19
B25	2939	3497	7	3515	7	3498	9	4.84	41	76.80	20
B26	2620	3593	5	3602	5	3601	5	1.88	11	88.64	4
B27	3504	3374	26	3405	27	3391	24	2.61	16	79.60	16
B28	3464	3417	18	3459	16	3416	19	7.97	57	65.87	40
B29	3435	3181	40	3233	39	3170	45	6.90	51	60.15	50
B3	3800	2970	60	2994	60	2985	60	2.09	13	65.83	41
B30	3072	3453	12	3471	14	3453	14	5.71	45	72.68	28
B31	3025	3171	43	3226	41	3169	46	8.14	58	56.61	55
B32	3225	2640	71	2690	70	2626	71	3.92	37	48.03	63
B33	3546	3397	22	3418	24	3395	22	5.72	46	70.60	31
B34	2857	2976	59	3133	53	2987	59	14.44	72	36.15	71
B35	3182	3033	55	3047	57	3045	55	1.33	6	69.83	33
B36	3190	2849	66	2906	63	2850	66	7.33	56	47.21	66
B37	3496	3425	17	3475	12	3423	18	9.03	64	63.35	42
B38	3143	3363	27	3410	26	3358	27	8.77	62	61.63	45
B39	3806	3180	41	3196	46	3190	40	1.74	9	74.09	25
B4	3487	2804	67	2828	67	2813	67	3.34	31	55.89	56
B40	3171	3447	14	3464	15	3461	13	1.52	8	84.77	8
B41	3006	2853	64	2898	64	2874	64	3.69	34	57.87	53
B42	2948	3136	50	3152	51	3148	49	2.58	15	70.17	32
B43	3110	2857	63	2892	65	2875	63	5.45	44	52.61	61
B44	3440	2919	61	2961	62	2941	61	3.55	33	60.67	49
B45	3228	2673	68	2695	69	2676	69	4.10	38	48.62	62
B46	3439	3720	3	3739	3	3734	3	3.21	29	90.21	2
B47	3411	3179	42	3214	45	3180	41	7.05	52	59.41	51
B48	3432	3162	45	3193	47	3173	43	4.51	40	66.05	39
B49	3476	2643	70	2685	71	2656	70	7.08	53	39.33	70
B5	3210	3305	31	3366	30	3316	30	8.71	61	60.82	48
B50	3489	2104	72	2192	72	2125	72	11.55	67	8.18	72
B51	3209	3485	9	3501	9	3498	8	1.75	10	85.50	5
B52	2586	3414	19	3432	20	3425	17	3.15	27	78.95	17
B53	3426	3451	13	3473	13	3466	12	2.86	21	81.36	13
B54	3129	3729	2	3765	1	3742	1	7.28	55	80.24	14
B55	2854	2657	69	2723	68	2683	68	6.56	47	42.72	68
B56	3491	3013	57	3066	56	3026	57	6.68	48	55.13	57
B57	3415	3209	36	3222	42	3220	37	1.19	3	76.64	21
B58	3201	3194	38	3221	43	3210	38	2.94	24	71.85	29
B59	3184	3161	46	3187	48	3172	44	4.48	39	66.07	38
B6	2594	3139	49	3177	50	3150	48	6.76	49	59.01	52
B60	3386	3386	24	3404	28	3387	25	2.77	20	78.54	18
B61	2031	3462	11	3477	11	3470	11	2.43	14	82.47	11
B62	3853	3403	20	3438	19	3403	21	7.18	54	67.32	36
B63	2652	3428	16	3441	18	3436	16	2.93	22	79.76	15
B64	3039	3204	37	3282	36	3208	39	9.82	65	54.28	58
B65	3221	3142	48	3252	38	3141	50	11.80	68	47.38	65
B66	3164	3401	21	3414	25	3411	20	1.07	2	84.06	10
B67	3414	3320	28	3427	22	3317	29	11.89	69	53.53	59
B68	3452	3732	1	3742	2	3738	2	2.72	19	91.39	1
B69	3283	2849	65	2889	66	2859	65	6.88	50	47.63	64
B7	3244	3274	34	3302	33	3292	32	3.71	36	72.74	27
B70	3438	3437	15	3455	17	3452	15	1.24	5	85.27	6
B71	2828	3164	44	3177	49	3176	42	1.20	4	75.00	22
B72	3319	3015	56	3029	58	3028	56	1.45	7	68.83	34
B8	3348	3192	39	3321	32	3221	36	8.23	60	60.92	47
B9	3219	3153	47	3218	44	3169	47	8.19	59	56.78	54

Supplemental Table S8: BLUP-based indexes for selecting genotypes with performance and stability at navy beans.

GEN	Y	HMGV	HMGV_R	RPGV	RPGV_R	HMRPGV	HMRPGV_R	WAASB	WAASB_R	WAASBY	WAASBY_R
N1	2469	2750	55	2765	55	2749	57	5.89	57	41.05	60
N10	2809	2934	33	2954	32	2950	32	2.32	21	67.63	24
N11	2734	2850	41	2905	39	2867	40	8.76	68	37.72	66
N12	2661	2534	69	2545	69	2542	69	3.19	33	40.23	62
N13	2849	2816	44	2827	46	2826	44	0.97	3	65.62	28
N14	2723	2739	57	2761	57	2752	55	3.01	31	53.58	43
N15	3302	2542	68	2567	68	2557	68	4.22	45	37.58	67
N16	3144	2595	67	2606	67	2605	67	1.85	18	49.36	51
N17	2677	2717	59	2731	59	2723	59	3.73	42	48.49	53
N18	3037	2679	61	2700	61	2693	61	2.05	19	54.08	41
N19	3090	2435	70	2456	70	2430	70	7.71	66	15.59	71
N2	2802	2668	62	2686	62	2682	62	3.52	39	46.01	56
N20	2913	2601	65	2613	66	2610	66	2.61	25	45.12	57
N21	2993	2965	30	2979	30	2977	28	1.45	8	72.39	14
N22	3119	3394	1	3417	1	3408	1	6.10	60	78.28	9
N23	2903	3104	11	3116	12	3115	11	1.71	14	79.06	7
N24	2852	2812	47	2824	47	2821	47	1.57	12	62.98	34
N25	2803	2790	51	2816	50	2808	49	3.75	43	53.92	42
N26	2969	2835	43	2855	43	2851	42	2.60	24	60.74	37
N27	2994	3059	19	3072	20	3066	19	2.63	26	72.57	13
N28	2805	2373	71	2401	71	2390	71	5.51	54	20.71	70
N29	2987	2935	32	2947	33	2946	33	1.42	7	70.28	19
N3	3337	3134	8	3152	8	3150	8	2.75	27	77.15	10
N30	3257	2813	45	2850	44	2823	45	7.49	65	39.31	65
N31	2632	2740	56	2755	58	2750	56	3.13	32	52.54	45
N32	2919	2598	66	2661	65	2626	65	6.51	63	32.84	69
N33	2630	2971	29	2981	29	2976	29	2.95	30	65.88	26
N34	2503	2985	26	3068	21	3016	23	8.94	69	47.46	54
N35	2544	3070	17	3115	13	3060	21	10.51	71	41.41	59
N36	2690	3229	3	3243	3	3233	4	5.13	49	71.80	17
N37	3207	3102	12	3116	11	3101	14	4.67	47	66.20	25
N38	3112	2708	60	2730	60	2704	60	6.28	62	37.31	68
N39	3035	3005	23	3016	25	3015	24	1.55	11	74.05	12
N4	3523	3069	18	3081	18	3079	17	0.94	2	80.27	5
N40	3156	3111	9	3125	10	3115	10	4.93	48	65.87	27
N41	3020	3056	20	3066	22	3061	20	3.23	34	69.60	20
N42	3312	2758	54	2769	54	2760	54	3.51	38	51.37	46
N43	2920	2795	49	2811	52	2799	51	4.52	46	49.71	50
N44	3156	2921	34	2931	35	2930	34	1.18	4	70.62	18
N45	2805	2785	52	2821	49	2784	53	6.72	64	40.89	61
N46	3114	2807	48	2832	45	2819	48	3.23	35	55.95	39
N47	2933	3210	5	3225	6	3222	6	2.15	20	83.58	3
N48	3290	3208	6	3252	2	3234	3	6.22	61	68.95	22
N49	3159	3161	7	3190	7	3181	7	5.23	50	69.26	21
N5	3043	3101	13	3115	14	3114	12	1.67	13	79.38	6
N50	3138	2661	63	2679	63	2671	63	3.37	36	47.24	55
N51	3293	2734	58	2761	56	2748	58	5.34	52	43.09	58
N52	2721	2792	50	2814	51	2808	50	3.64	40	54.21	40
N53	2901	2856	39	2901	40	2879	39	6.08	59	49.35	52
N54	2344	3095	14	3112	15	3109	13	1.84	17	78.50	8
N55	2619	3109	10	3136	9	3124	9	5.76	56	64.00	31
N56	2521	2918	35	2939	34	2916	35	6.04	58	50.31	49
N57	3100	3056	21	3074	19	3071	18	2.87	29	72.14	16
N58	2261	2953	31	2975	31	2969	30	3.45	37	64.10	30
N59	2947	2990	24	3006	28	3004	25	1.82	16	72.37	15
N6	2824	3017	22	3025	23	3023	22	1.40	6	74.94	11
N60	2602	3091	15	3101	16	3101	15	0.86	1	81.68	4
N61	3122	2973	28	3018	24	2965	31	9.45	70	39.86	63
N62	3132	2851	40	2883	41	2846	43	7.84	67	39.48	64
N63	3160	2871	38	2907	38	2893	38	5.28	51	52.99	44
N64	2731	3078	16	3090	17	3082	16	3.96	44	67.96	23
N65	2784	2891	37	2912	37	2894	37	5.71	55	50.44	48
N66	2790	2990	25	3008	27	2994	27	5.50	53	56.89	38
N67	2803	2655	64	2675	64	2669	64	2.48	22	50.87	47
N68	2716	2984	27	3008	26	3001	26	3.67	41	65.18	29
N69	3192	3230	2	3243	4	3242	2	1.78	15	86.14	2
N7	2930	3218	4	3229	5	3229	5	1.26	5	87.42	1
N70	3025	2894	36	2914	36	2909	36	2.83	28	63.18	33
N71	3007	2781	53	2795	53	2794	52	1.50	10	60.89	36
N8	2859	2842	42	2860	42	2854	41	2.52	23	61.21	35
N9	2755	2812	46	2823	48	2821	46	1.47	9	63.27	32

Supplemental Table S9: BLUP-based indexes for selecting genotypes with performance and stability at red beans.

GEN	Y	HMGV	HMGV_R	RPGV	RPGV_R	HMRPGV	HMRPGV_R	WAASB	WAASB_R	WAASBY	WAASBY_R
R1	3213	3307	5	3332	5	3331	5	3.11	3	70.44	3
R10	2995	3174	8	3206	8	3193	7	5.37	10	54.74	7
R11	3170	3181	7	3212	7	3183	8	7.65	16	47.05	11
R12	3420	3005	13	3027	14	3014	14	5.44	12	41.75	14
R13	3020	3155	9	3180	9	3163	9	5.11	9	53.00	8
R14	2762	3681	1	3724	1	3679	1	11.37	19	69.93	4
R15	3083	2990	15	3012	15	3001	16	5.42	11	40.95	16
R16	2903	2987	16	3005	16	3003	15	1.54	1	52.65	9
R17	3363	3006	12	3096	10	3014	13	14.62	21	18.25	20
R18	2835	3323	4	3370	4	3360	4	3.70	6	71.93	2
R19	3040	3030	10	3080	11	3063	10	5.49	13	46.71	12
R2	2956	3233	6	3258	6	3251	6	4.06	7	62.28	6
R20	2981	2819	20	2939	18	2873	19	12.06	20	17.67	21
R21	3274	3420	3	3445	3	3444	3	3.14	4	78.03	1
R3	2978	2850	19	2875	20	2872	20	3.50	5	38.31	17
R4	3494	3018	11	3040	13	3036	11	2.56	2	52.00	10
R5	3201	3003	14	3047	12	3031	12	6.31	14	41.72	15
R6	3786	3474	2	3504	2	3482	2	8.32	18	65.20	5
R7	3073	2722	21	2794	21	2765	21	7.67	17	21.26	19
R8	2930	2933	17	2972	17	2965	17	4.22	8	43.20	13
R9	3546	2860	18	2919	19	2900	18	6.72	15	32.45	18

Supplemental Table S10: Selection differential of the WAASBY index for 27 Black, 29 Navy and 12 Red beans.

Black Beans

Factor	Trait	Xo [‡]	Xs [§]	SD (%) [¶]	Sense
FA 1	GY	3,201	3,258	57.92 (1.81)	Increase
FA 1	LD	1.73	1.63	-0.1 (-5.556)	Decrease
FA 2	DM	91.78	91.90	0.12 (0.1324)	Decrease
FA 2	PH	18.15	18.34	0.19 (1.063)	Increase

Navy Beans

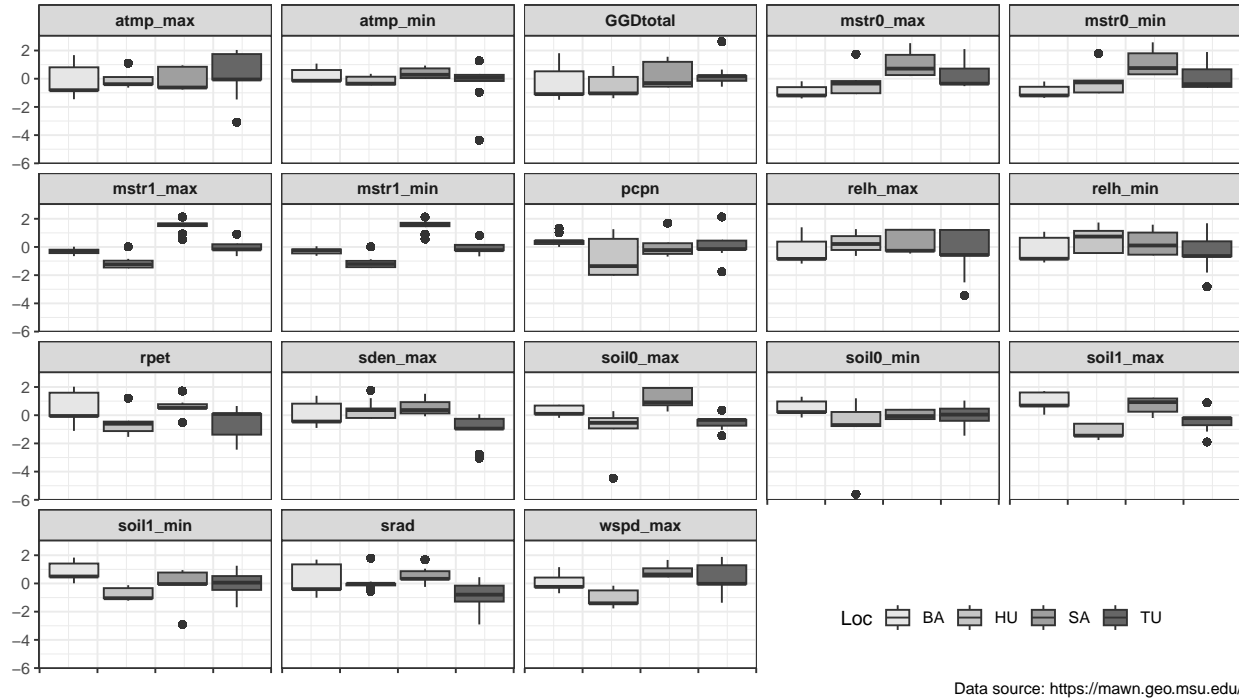
Factor	Trait	Xo [‡]	Xs [§]	SD (%) [¶]	Sense
FA 1	GY	3,006	3,145	138.90 (4.619)	Increase
FA 1	DM	91.25	91.54	0.29 (0.3184)	Decrease
FA 1	PH	18.34	19.69	1.35 (7.371)	Increase
FA 1	LD	1.88	1.65	-0.22 (-11.85)	Decrease

Red Beans

Factor	Trait	Xo [‡]	Xs [§]	SD (%) [¶]	Sense
FA 1	PH	18.25	20.07	1.82 (9.966)	Increase
FA 1	LD	1.87	1.91	0.04 (2.104)	Decrease
FA 2	GY	3,156	3,162	5.92 (0.1875)	Increase
FA 2	DM	90.85	90.88	0.03 (0.02776)	Decrease

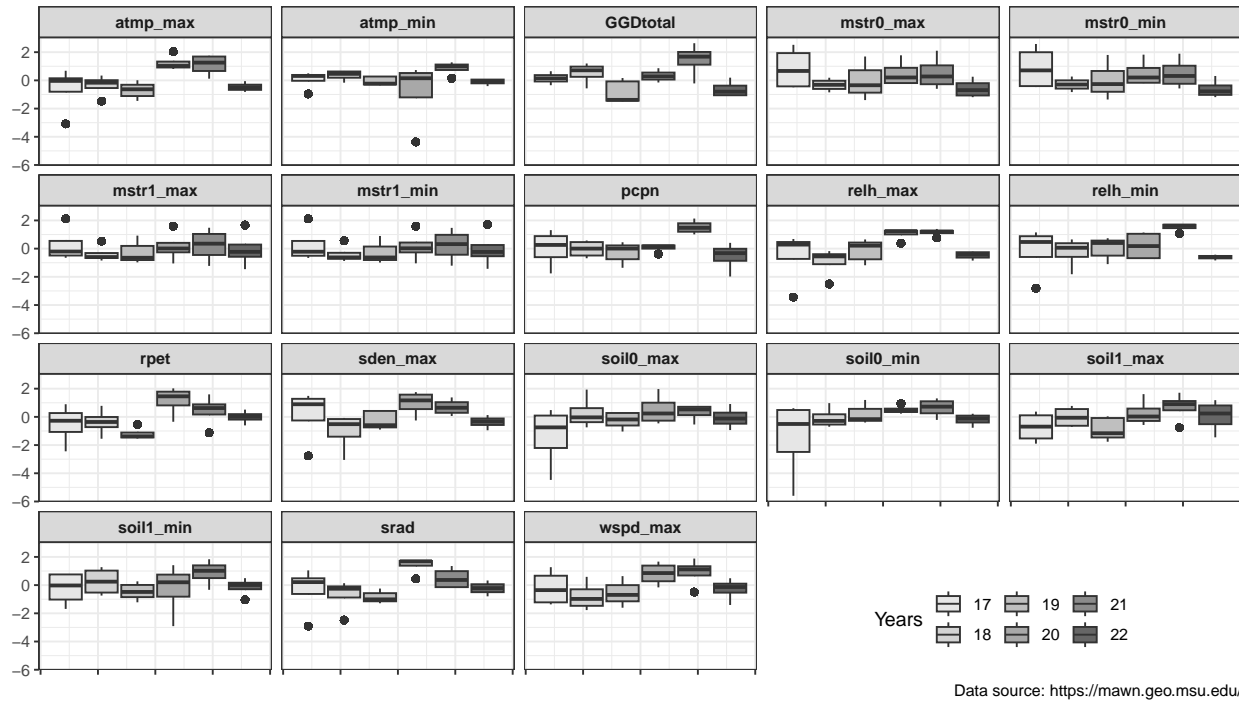
GY, grain yield; PH, plant height; DM, days to maturity; LD, lodging;[‡] Xo, mean for WAASBY index of the original population; [§] Xs, mean for WAASBY index of the selected genotypes (BB: B55, B1, B29, B20, B28; NB: N38, N6, N61, N35, N52, N22; SR: R2, R13); SD, Selection differential; [¶]% of SD in parenthesis.

2 Appendix B - Supplemental Figures



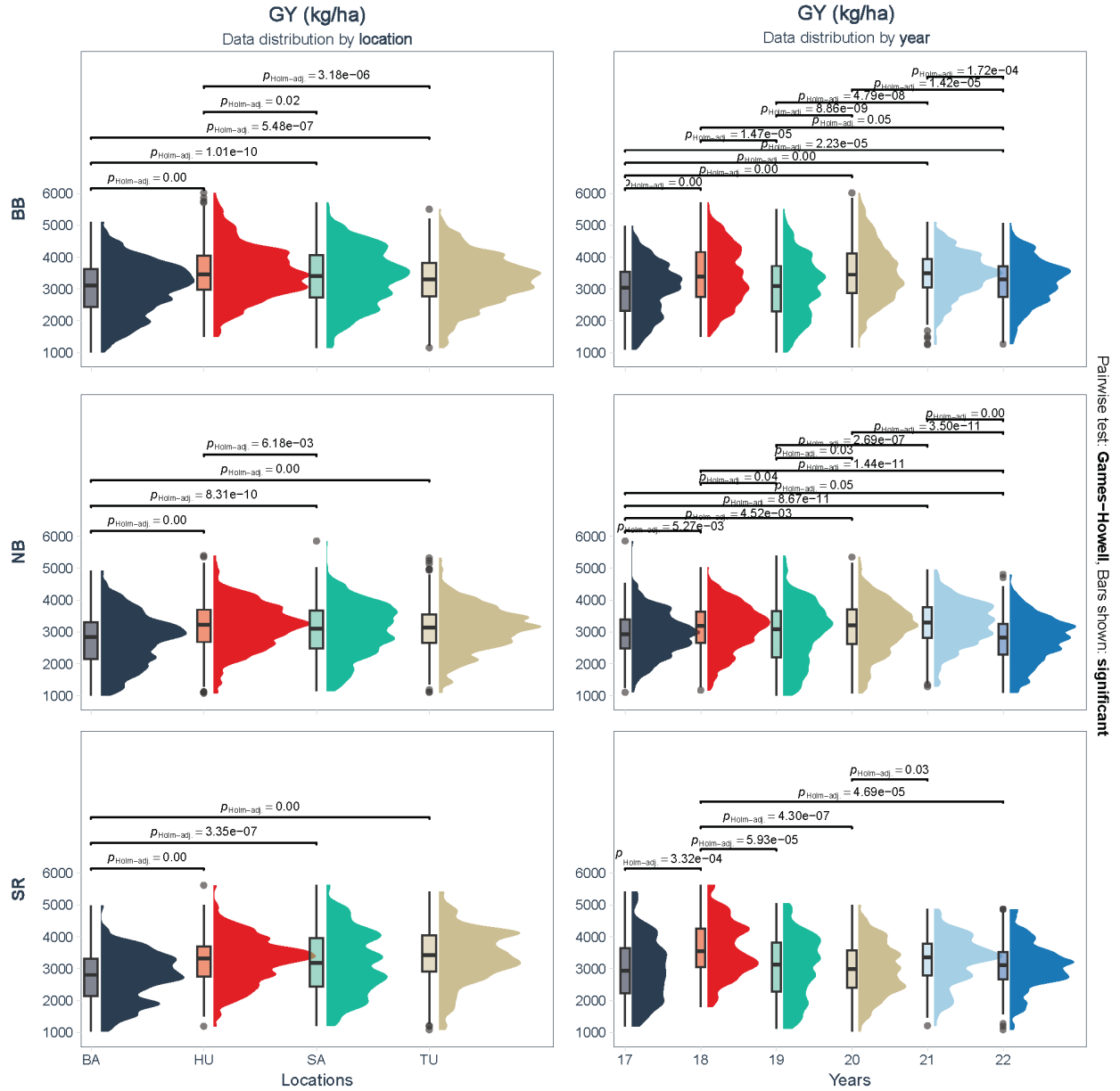
Supplemental Figure S1: Box plot distribution from the weather data collected by location from 2017 to 2022 cultivation years. Data were scaled to better plot visualization. Weather data obtained using daily average value from planting to harvesting. TU: Tuscola, SA: Sanilac, HU: Huron, and BA: Bay locations.

atmp_max: Max Air Temperature (1.5m), atmp_min: Min Air Temperature (1.5m), mstr0_max: Max Soil Moisture (0-30cm), mstr0_min: Min Soil Moisture (0-30cm), mstr1_max: Max Soil Moisture (30-60cm), mstr1_min: Min Soil Moisture (30-60cm), pcpn: Precipitation, relh_max: Max Relative Humidity (1.5m), relh_min: Min Relative Humidity (1.5m), rpet: Reference Potential Evapotranspiration, sden_max: Max Solar Flux, soil0_max: Max Soil Temperature (5cm), soil0_min: Min Soil Temperature (5cm), soil1_max: Max Soil Temperature (10cm), soil1_min: Min Soil Temperature (10cm), srad: Total Solar Flux, wspd_max: Max Wind Speed (3m). GGDtotal: Growing Degree-Day Calculations in Celsius (C) given by: $(max - min)/2 - 10$

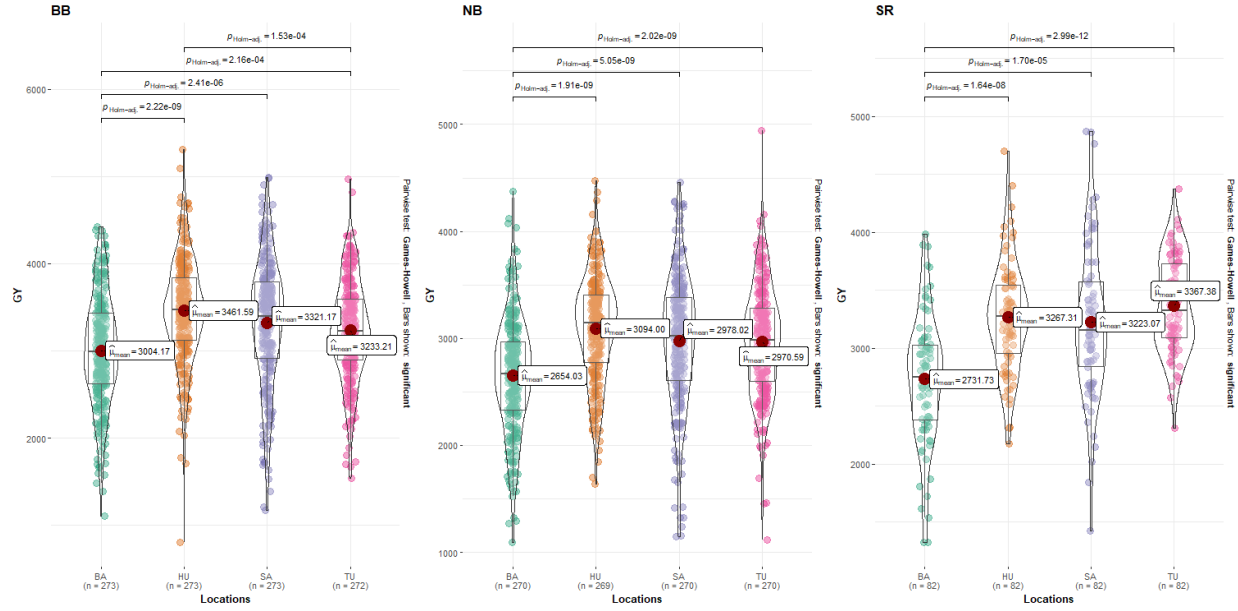


Supplemental Figure S2: Box plot distribution from the weather data collected by year from 2017 to 2022 cultivation locations. Data were scaled to better plot visualization. Weather data obtained using daily average value from planting to harvesting. TU: Tuscola, SA: Sanilac, HU: Huron, and BA: Bay locations.

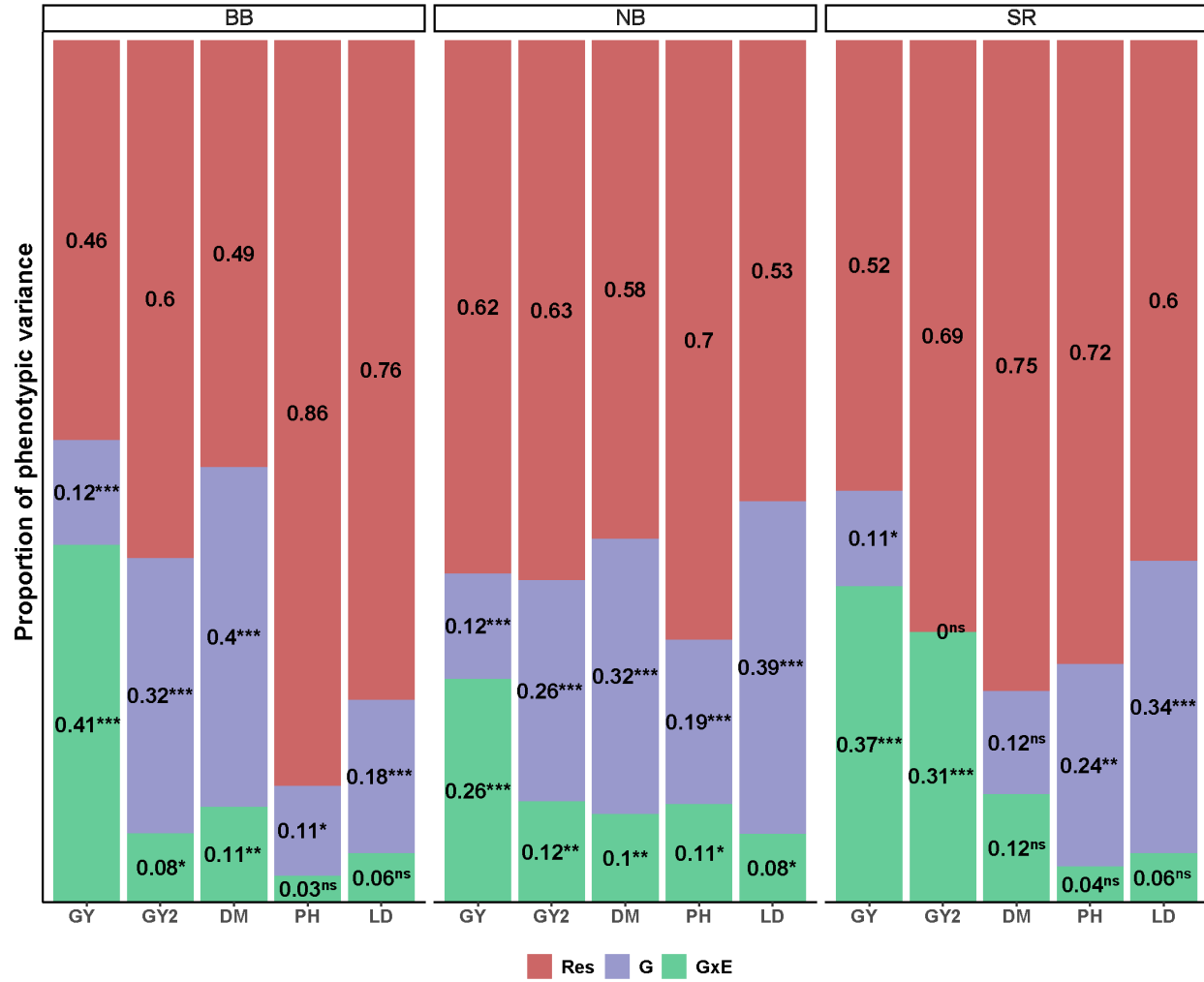
atmp_max: Max Air Temperature (1.5m), atmp_min: Min Air Temperature (1.5m), mstr0_max: Max Soil Moisture (0-30cm), mstr0_min: Min Soil Moisture (0-30cm), mstr1_max: Max Soil Moisture (30-60cm), mstr1_min: Min Soil Moisture (30-60cm), pcpn: Precipitation, relh_max: Max Relative Humidity (1.5m), relh_min: Min Relative Humidity (1.5m), rpet: Reference Potential Evapotranspiration, sden_max: Max Solar Flux, soil0_max: Max Soil Temperature (5cm), soil0_min: Min Soil Temperature (5cm), soil1_max: Max Soil Temperature (10cm), soil1_min: Min Soil Temperature (10cm), srاد: Total Solar Flux, wspd_max: Max Wind Speed (3m). GGDtotal: Growing Degree-Day Calculations in Celsius (C) given by: $(max - min)/2 - 10$



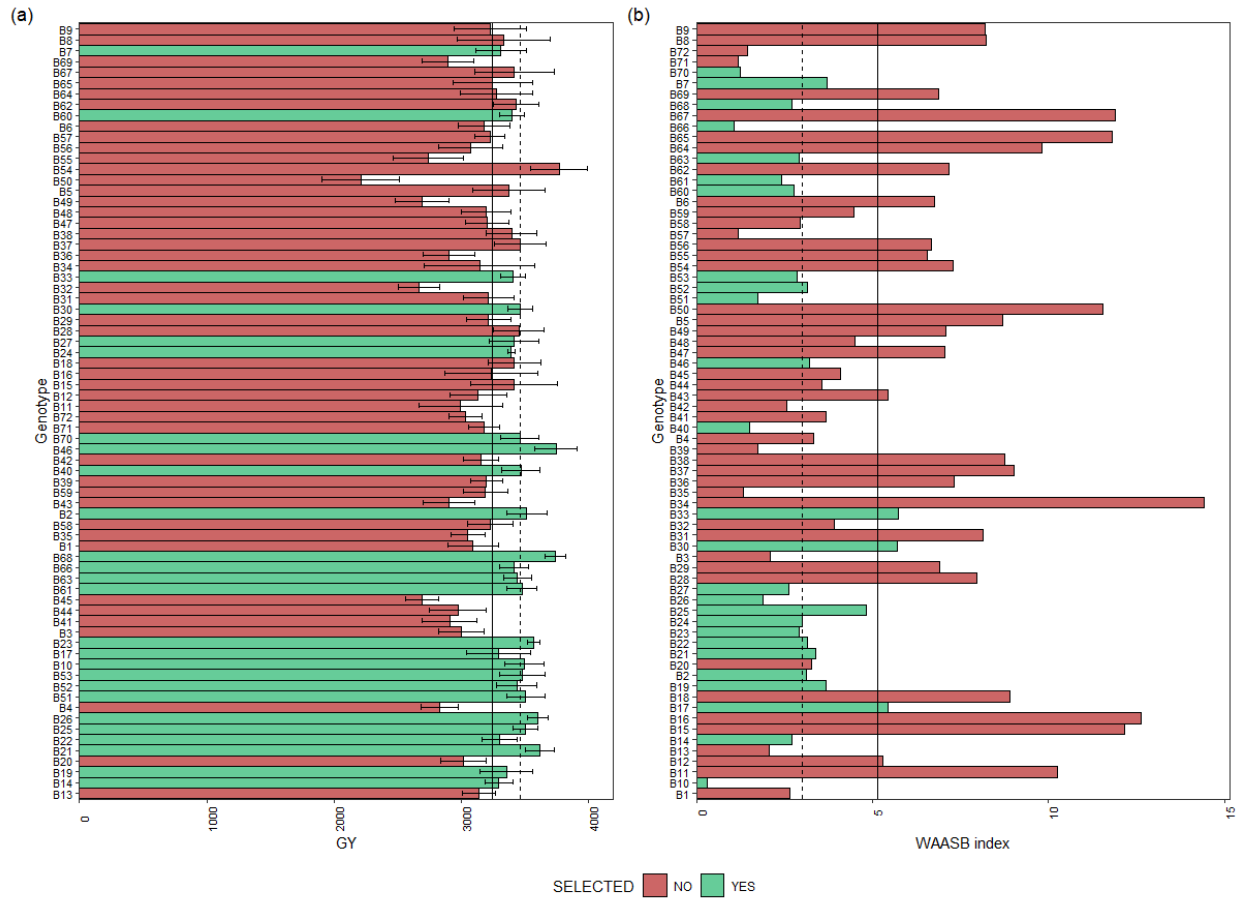
Supplemental Figure S3: Combination of box and density plots for between subjects comparisons by locations (left) and year (right) of grain yield (GY in Kg/ha) for black (BB), Navy (NB) and Small Red (SR) beans. BA: Bay, HU: Huron, SA: Sanilac, TU: Tuscola locations.



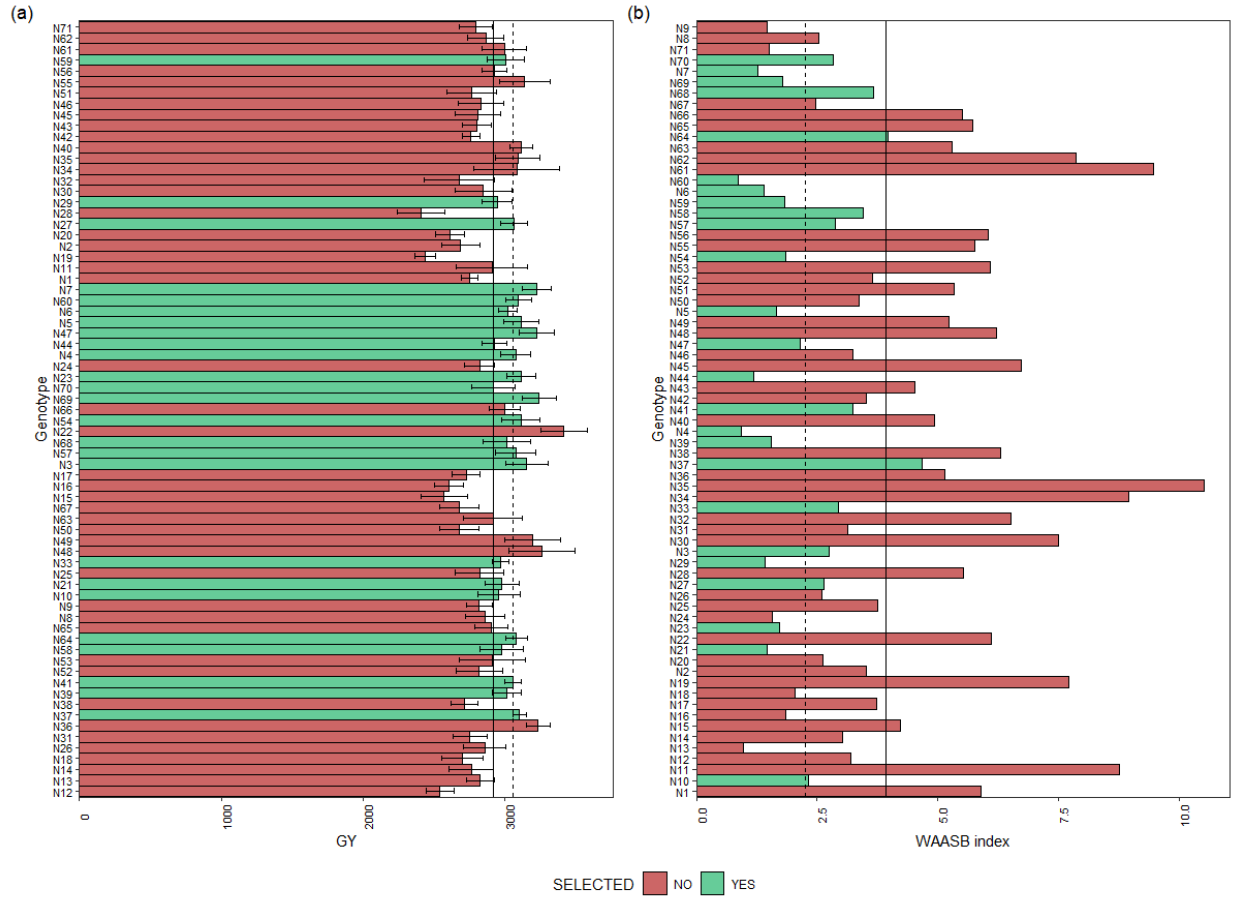
Supplemental Figure S4: Combination of box and violin plots along with jittered data points and grand mean values for between subjects comparisons by locations of grain yield (GY) for black (BB), Navy (NB) and Small Red (SR) beans. Pairwise Games-Howell test used. Comparisons showing only significant values. BA: Bay, HU: Huron, SA: Sanilac, TU: Tuscola. Pairwise Games-Howell test used. Comparisons showing only significant between the pairs of environments



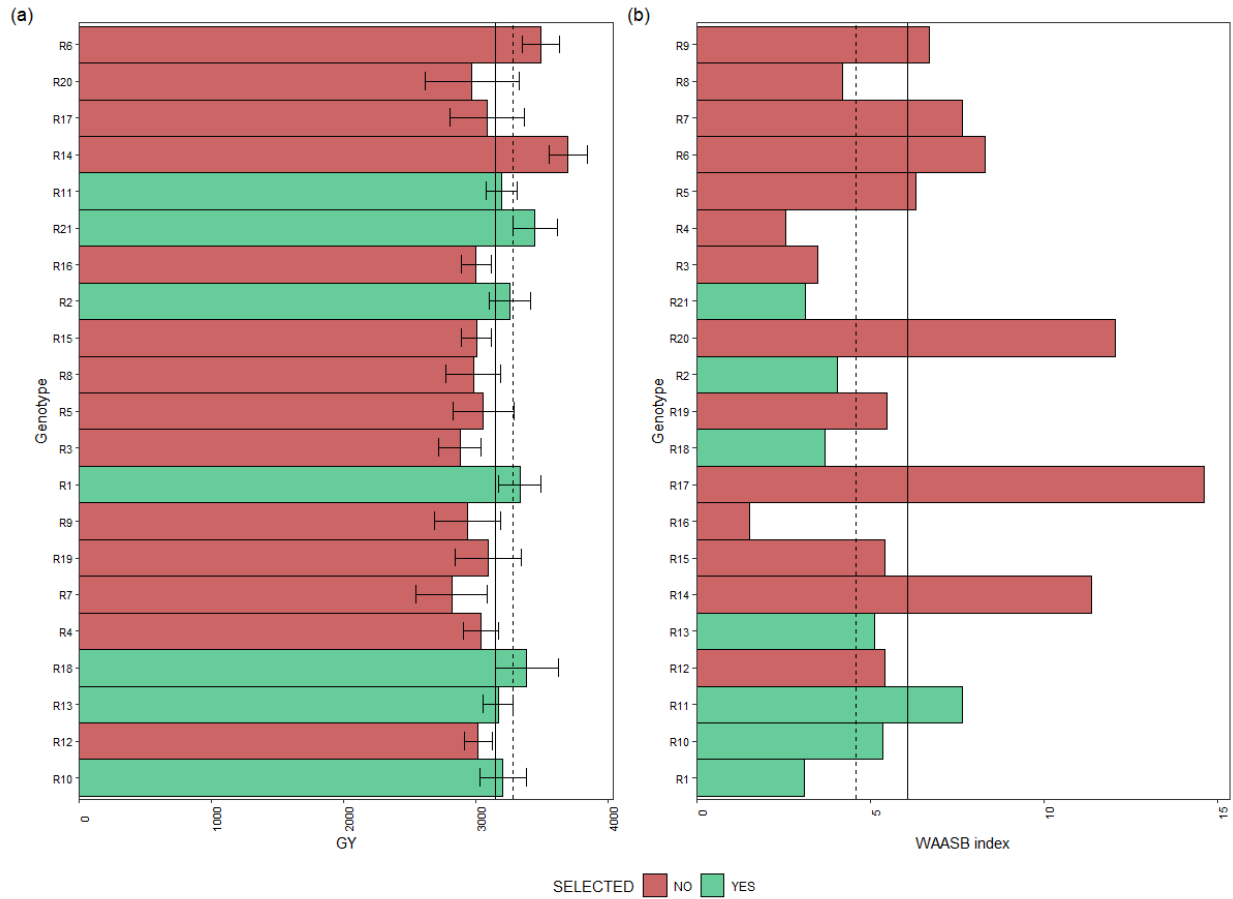
Supplemental Figure S5: Proportion of the phenotypic variance for 71 Black (BB), 72 Navy (NB), and 21 Red (SR) beans for grain yield in Kg/ha (GY: 2017 - 2022; GY2: only 2021) days to maturity (DM, days), plant height (PH, cm) and lodging (LD, scale) traits evaluated in the study. * Significant at $P < 0.05$. ** Significant at $P < 0.01$. *** Significant at $P < 0.001$. ns, nonsignificant.



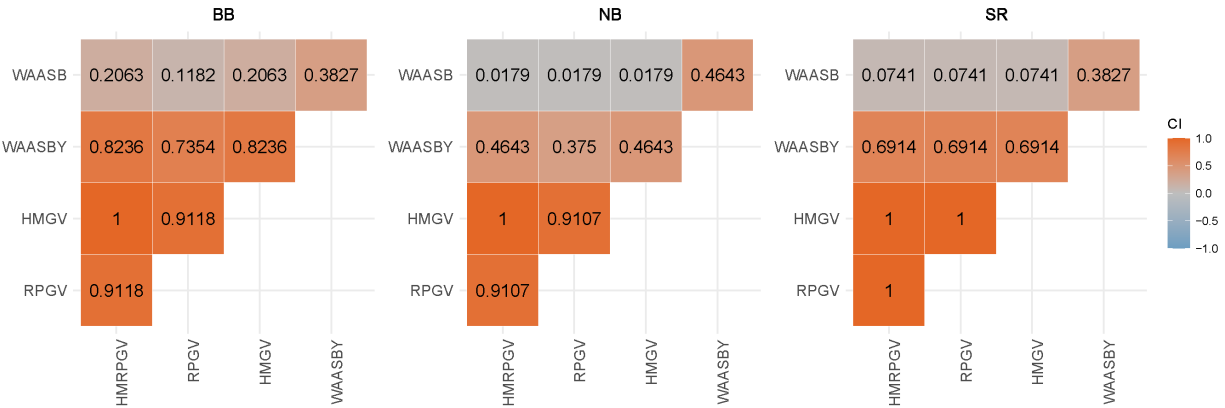
Supplemental Figure S7: Mean performance (a) and stability (b) for grain yield (GY) of 72 black beans genotypes. The vertical dashed and solid lines shows, respectively, the mean of the selected genotype and the overall mean for both mean performance and WAASB index.



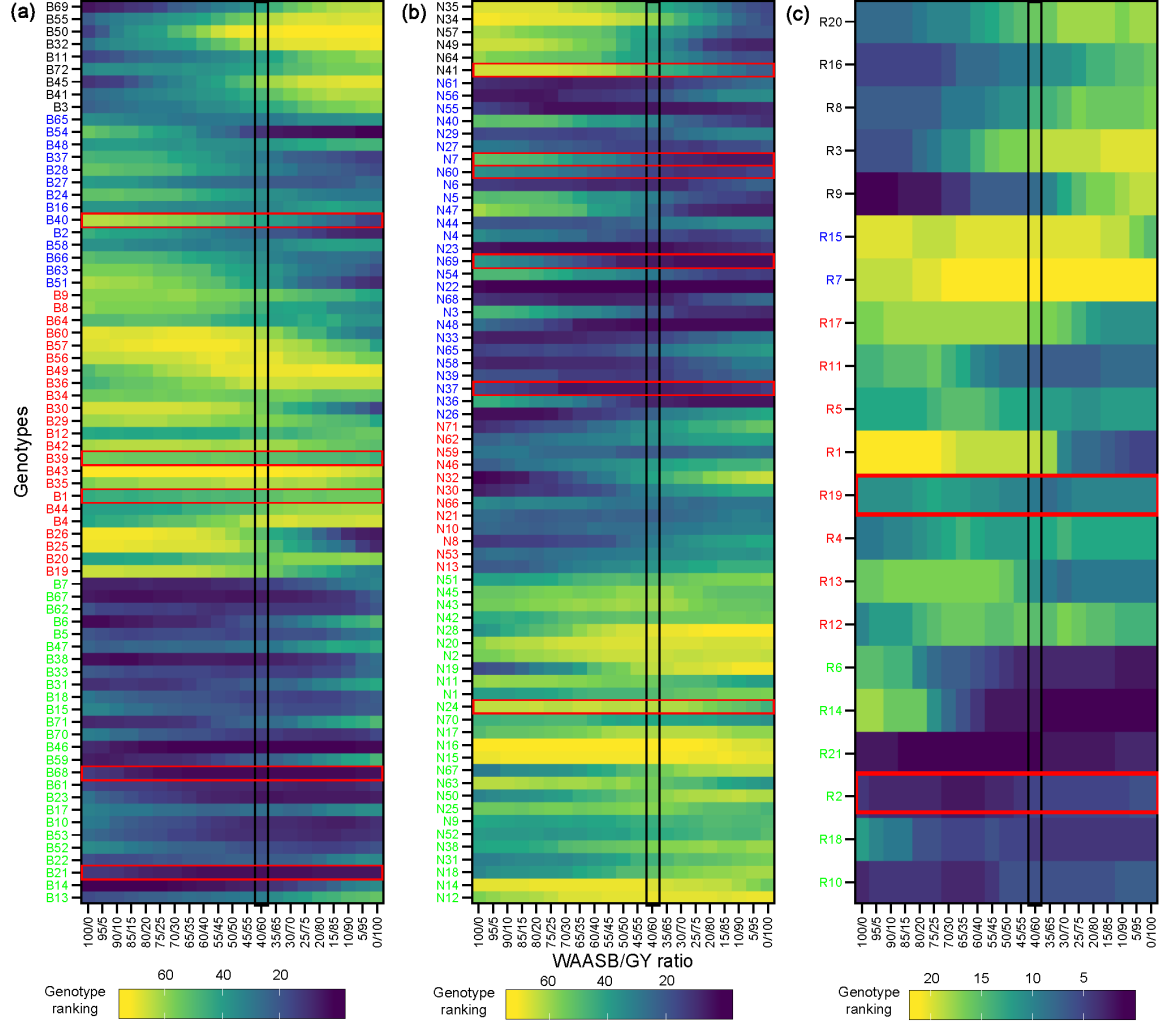
Supplemental Figure S8: Mean performance (a) and stability (b) for grain yield (GY) of 71 navy beans genotypes. The vertical dashed and solid lines shows, respectively, the mean of the selected genotype and the overall mean for both mean performance and WAASB index.



Supplemental Figure S9: Mean performance (a) and stability (b) for grain yield (GY) of 21 red beans genotypes. The vertical dashed and solid lines shows, respectively, the mean of the selected genotype and the overall mean for both mean performance and WAASB index.



Supplemental Figure S10: BLUP-based stability indexes coincidence index (CI) using selection intensity of 20 top genotypes. BB, black; NB, navy; SR, red beans.



Supplemental Figure S11: Ranks of dry beans genotypes (a: 72 black, b:71 navys and c: 21 small red/pink) considering different weights for stability and yielding. The most-left ranks were obtained considering the stability only. The most right-ranks were obtained considering the grain yield only. Between the extremes, the ranks were obtained different weights for stability and yielding. The four clusters represent four classes of genotypes: (1) Poorly productive and unstable genotypes; (2) productive but unstable genotypes; (3) stable but poorly productive genotypes; and (4), highly productive and stable genotypes. The ranks highlighted by a black rectangle are the same from those BLUPs predicted to WAASBY index and the red rectangle box are the selected genotypes by the Multi-Trait Stability Index (MTSI).

3 Appendix C - R codes

Getting started

The present analysis aims to dissect the genotype by environment interaction study (aka GEI) using a data set from the Dry Beans breeding program at Michigan State University - MSU.

The trait in study is the grain yield (GY) per plot (lb/plot) adjusted to the international measurements (Kg/ha). A previous data analysis (not shown here) was done to perform the historical data mining and adjust of raw data. Besides the GY, plant height (PH), date of maturity (DM) and lodging (LD) were investigate using a subset from 2021, which contains the all the data available.

The main focus of this manuscript, as describe in the published paper, is to investigate the varieties performance of GY across four locations at different Michigan counties. However, the MTME (three locations and 4 traits) also was studied in this manuscript when available (only in 2021 in Bay, Tuscola and Sanilac sites). Thus, different types of analysis will be performed in order to study the GEI in the Multi-Environment-Trials (MET) data to provide better varieties recommendations to the Dry Beans breeding program, which:

- ☒ Multi-Environment Trials – Genotype x Environment Interaction to grain yield (GY)
- ☒ Mean performance and stability of multiple traits

3.1 Packages

This document was prepared on 2023-05-22.

The analysis was done using the R Statistical language (v4.2.2; R Core Team, 2022) on Windows 10 x64, using the packages rmarkdown (v2.19), ggpmisc (v0.5.1), ggpp (v0.5.0), gridExtra (v2.3), magrittr (v2.0.3), Matrix (v1.5.1), mapdata (v2.3.1), maps (v3.4.1), spData (v2.2.1), asremlPlus (v4.3.40), asreml (v4.1.0.160), tidyquant (v1.0.6), data.table (v1.14.4), flextable (v0.8.3), lubridate (v1.9.0), purrr (v0.3.5), raster (v3.6.3), ggcorrplot (v0.1.4), ggdist (v3.2.1), tibble (v3.1.8), rcartocolor (v2.0.0), metan (v1.17.0), ggstatsplot (v0.10.0), sf (v1.0.9), sp (v1.5.1), ggforce (v0.4.1), PerformanceAnalytics (v2.0.4), broom (v1.0.2), quantmod (v0.4.20), xts (v0.12.2), openxlsx (v4.2.5.1), timechange (v0.1.1), naniar (v0.6.1), TTR (v0.24.3), statgenGxE (v1.0.5), tigris (v2.0), plyr (v1.8.8), ggplot2 (v3.4.0), forcats (v0.5.2), stringr (v1.5.0), tidyverse (v1.3.2), dplyr (v1.0.10), tidyr (v1.2.1), readr (v2.1.3), cowplot (v1.1.1), nadiv (v2.17.2), DT (v0.26), zoo (v1.8.11) and kableExtra (v1.3.4).

3.2 Data preparation

```
data_beans = read.csv("data/DataBean_MET_GYv2.csv",h=T, stringsAsFactors = T)

if (knitr::is_html_output()) {

  print_table(data_beans)

}else{

  flextable(head(data_beans)) %>%
    add_footer_lines(
      c("Varieties Dry Beans data set from 2017 to 2022",
        "Header data set showing the 6 first entry")) %>%
    autofit() %>%
```

```
add_header_lines("Dry Beans varieties trial") %>%
theme_design2()
}
```

Dry Beans varieties trial

codename	name	rep	mkt	year_loc	loc	year	gy_kg_ha
213SP	N1	1NB	17_BA	BA		17	
213SP	N1	1NB	17_HU	HU		17	
213SP	N1	1NB	17_SA	SA		17	
213SP	N1	1NB	17_TU	TU		17	
213SP	N1	1NB	18_BA	BA		18	
213SP	N1	1NB	18_HU	HU		18	

Varieties Dry Beans data set from 2017 to 2022

Header data set showing the 6 first entry

```
# Data adjustment
# All the effect columns must be as a factor to run in ASReml-r.
cols <- c("rep", "name", "loc", "year", "mkt", "year_loc")
data_beans[cols] <- lapply(data_beans[cols], factor)
data_beans <- data.table(data_beans)
```

3.3 Descriptive Stats - Raw data

Data set distribution, checking data and locations of study.

```
data_beans = read.csv("data/DataBean_MET_GYv2.csv", h=T, stringsAsFactors = T)
```

```
# Data adjustment
# All the effect columns must be as a factor to run in ASReml-r.
cols <- c("rep", "name", "loc", "year", "mkt", "year_loc")
data_beans[cols] <- lapply(data_beans[cols], factor)
data_beans <- data.table(data_beans)
```

3.3.1 Box plot dist.

```
plotDM_stats_mkt_loc <- grouped_ggbetweenstats(data=data_beans, x= loc, y=gy_kg_ha, type =
  ↪ "parametric", bf.message = F, results.subtitle = F,
  ylab= "GY", xlab = "Locations",
  plot.type = "boxviolin", grouping.var = mkt )
```

```

#print(plotDM_stats_mkt_loc)

plotDM_stats_mkt_year<- grouped_ggbetweenstats(data=data_beans, x= year, y=gy_kg_ha, type
  ↪ = "parametric", bf.message = F, results.subtitle = F,
      ylab= "GY", xlab = "Years",
      plot.type = "boxviolin", grouping.var = mkt )

#print(plotDM_stats_mkt_year)

print(arrange_ggplot(plotDM_stats_mkt_loc,plotDM_stats_mkt_year))

a<- ggplot(data=data_beans, aes(x=reorder(loc, -gy_kg_ha), y=gy_kg_ha, fill=loc)) +
  ggdist::stat_halfeye(
    adjust = 0.5,
    justification = -0.1,
    .width = 0,
    point_colour = NA
  ) +

  geom_boxplot(
    width = .12,
    # outlier.color = NA,
    alpha = 0.5)+
  tidyquant::theme_tq()+
  facet_wrap("year")+
  facet_grid("mkt") +
  tidyquant::scale_fill_tq() +
  scale_x_discrete(expand = c(0.03,0)) +
  scale_y_continuous(limits = c(1000,6000), breaks = seq(1000, 6000, by = 1000)) +
  theme(axis.text.x=element_text(angle = 90),
    strip.text=element_blank(),
    legend.position = "none",
    panel.grid = element_blank())+
  labs(title="GY (kg/ha)",
    subtitle="Data distribution by location",
    caption=NULL, x=NULL, y=NULL)

b<- ggplot(data=data_beans, aes(x=reorder(year, -gy_kg_ha), y=gy_kg_ha, fill=year)) +
  ggdist::stat_halfeye(
    adjust = 0.5,
    justification = -0.1,
    .width = 0,
    point_colour = NA
  ) +

  geom_boxplot(
    width = .12,
    #outlier.color = NA,
    alpha = 0.5)+
  tidyquant::theme_tq()+
  facet_wrap("loc")+
  facet_grid("mkt") +
  tidyquant::scale_fill_tq() +
  scale_x_discrete(expand = c(0.03,0)) +

```

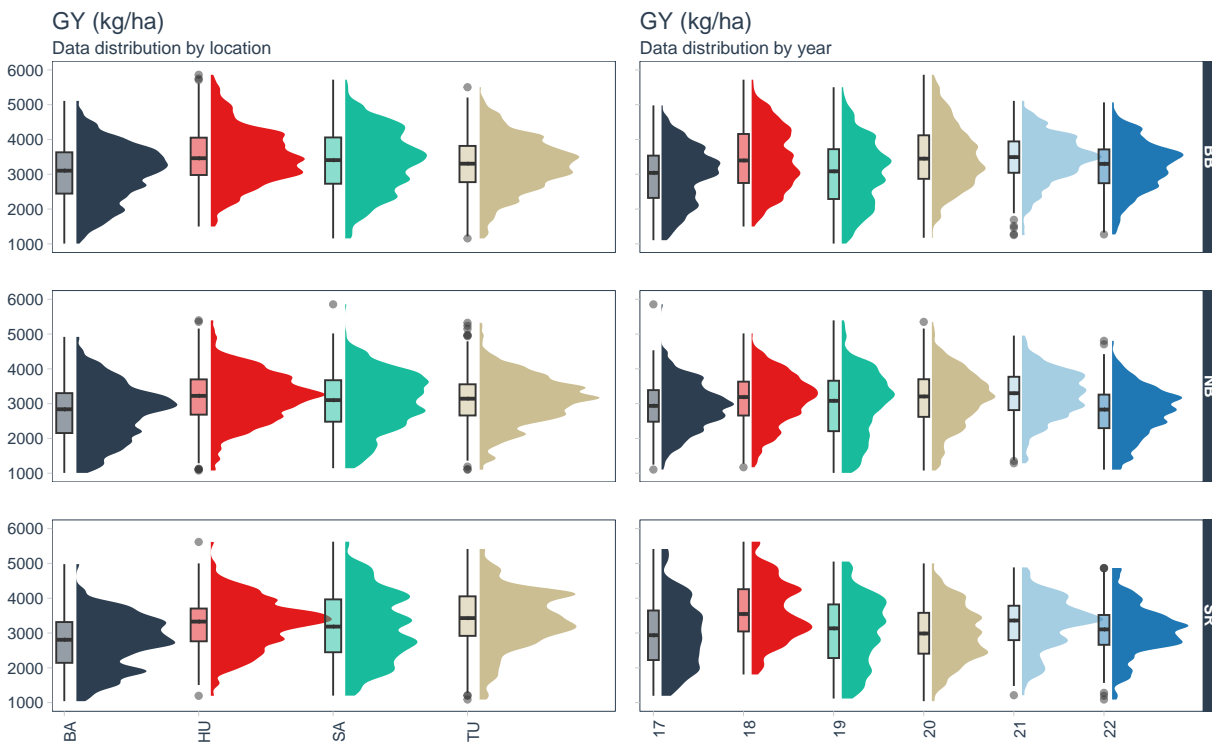


```

scale_y_continuous(limits = c(1000,6000), breaks = seq(1000, 6000, by = 1000)) +
theme(axis.text.x=element_text(angle = 90),
      strip.text=element_text(face="bold"),
      legend.position = "none",
      axis.text.y = element_blank(),
      panel.grid = element_blank())+
labs(title="GY (kg/ha)",
      subtitle="Data distribution by year",
      caption=NULL, x=NULL, y=NULL)

print(arrange_ggplot(a,b))

```



```

data_beans_plotBB<- droplevels(subset(data_beans, mkt=="BB"))

stats_results_BB<- ggbetweenstats(data=data_beans_plotBB, x= loc, y=gy_kg_ha) %>%
  ↪ extract_stats()
#print(stats_results_BB$pairwise_comparisons_data$p.value)

stats_results_BB2<- stats_results_BB$pairwise_comparisons_data %>%
dplyr::mutate(groups = purrr::pmap(.l = list(group1, group2), .f = c)) %>%
  dplyr::arrange(group1)%>%
  filter(p.value <= 0.05)

bb1<- ggplot(data=data_beans_plotBB, aes(x=reorder(loc, -gy_kg_ha), y=gy_kg_ha,
  ↪ fill=loc)) +

```

```

ggdist::stat_halfeye(
  adjust = 0.5,
  justification = -0.1,
  .width = 0,
  point_colour = NA
) +

geom_boxplot(
  width = .12,
  # outlier.color = NA,
  alpha = 0.5)+
tidyquant::theme_tq()+
#facet_grid("loc")+
tidyquant::scale_fill_tq() +
scale_x_discrete(expand = c(0.03,0)) +
scale_y_continuous(limits = c(1000,10000), breaks = seq(1000, 10000, by = 1000)) +
theme(axis.text.x=element_blank(),
      strip.text=element_blank(),
      legend.position = "none",
      panel.grid = element_blank())+
labs(title="GY (kg/ha)",
      subtitle="Data distribution by location",
      caption=NULL, x=NULL, y="BB")

bb1<-bb1 +
ggsignif::geom_signif(
  comparisons = stats_results_BB2$groups,
  map_signif_level = TRUE,
  tip_length = 0.01,
  textsize = 2.7,
  y_position = c(5900, 6700, 7500,8300, 9100),
  annotations = as.character(stats_results_BB2$expression),
  test = NULL,
  na.rm = TRUE,
  parse = TRUE
)

```

```

data_beans_plotBB<- droplevels(subset(data_beans, mkt=="BB"))

stats_results_BB<- ggbetweenstats(data=data_beans_plotBB, x= year, y=gy_kg_ha) %>%
  ↪ extract_stats()
#print(stats_results_BB$pairwise_comparisons_data$p.value)

stats_results_BB3<- stats_results_BB$pairwise_comparisons_data %>%
dplyr::mutate(groups = purrr::pmap(.l = list(group1, group2), .f = c)) %>%
  dplyr::arrange(group1) %>%
  filter(p.value <= 0.05)

bb2<- ggplot(data=data_beans_plotBB, aes(x=reorder(year, -gy_kg_ha), y=gy_kg_ha,
  ↪ fill=year)) +
  ggdist::stat_halfeye(
    adjust = 0.5,

```

```

      justification = -0.1,
      .width = 0,
      point_colour = NA
    ) +

    geom_boxplot(
      width = .12,
      # outlier.color = NA,
      alpha = 0.5)+
    tidyquant::theme_tq()+
    #facet_grid("loc")+
    tidyquant::scale_fill_tq() +
    scale_x_discrete(expand = c(0.03,0)) +
    scale_y_continuous(limits = c(1000,10000), breaks = seq(1000, 10000, by = 1000)) +
    theme(axis.text.x=element_blank(),
          strip.text=element_blank(),
          legend.position = "none",
          panel.grid = element_blank())+
    labs(title="GY (kg/ha)",
         subtitle="Data distribution by year",
         caption=NULL, x=NULL, y=NULL)

bb2<-bb2 +
  ggsignif::geom_signif(
    comparisons = stats_results_BB3$groups,
    map_signif_level = TRUE,
    tip_length = 0.01,
    textsize = 2.7,
    y_position = c(5900, 6300, 6700,7100, 7400, 7800,8200, 8600, 9000, 9400),
    annotations = as.character(stats_results_BB3$expression),
    test = NULL,
    na.rm = TRUE,
    parse = TRUE
  )

```

3.3.1.1 BB

```

data_beans_plotBB<- droplevels(subset(data_beans, mkt=="NB"))

stats_results_BB<- ggbetweenstats(data=data_beans_plotBB, x= loc, y=gy_kg_ha) %>%
  ↪ extract_stats()
#print(stats_results_BB$pairwise_comparisons_data$p.value)

stats_results_BB2<- stats_results_BB$pairwise_comparisons_data %>%
dplyr::mutate(groups = purrr::pmap(.l = list(group1, group2), .f = c)) %>%
  dplyr::arrange(group1)%>%
  filter(p.value <= 0.05)

nb1<- ggplot(data=data_beans_plotBB, aes(x=reorder(loc, -gy_kg_ha), y=gy_kg_ha,
  ↪ fill=loc)) +

```

```

ggdist::stat_halfeye(
  adjust = 0.5,
  justification = -0.1,
  .width = 0,
  point_colour = NA
) +

geom_boxplot(
  width = .12,
  # outlier.color = NA,
  alpha = 0.5)+
tidyquant::theme_tq()+
#facet_grid("loc")+
tidyquant::scale_fill_tq() +
scale_x_discrete(expand = c(0.03,0)) +
scale_y_continuous(limits = c(1000,10000), breaks = seq(1000, 10000, by = 1000)) +
theme(axis.text.x=element_blank(),
      strip.text=element_blank(),
      legend.position = "none",
      panel.grid = element_blank())+
labs(
  caption=NULL, x=NULL, y="NB")

nb1<-nb1 +
ggsignif::geom_signif(
  comparisons = stats_results_BB2$groups,
  map_signif_level = TRUE,
  tip_length = 0.01,
  textsize = 2.7,
  y_position = c(5900, 6700, 7500,8300, 9100),
  annotations = as.character(stats_results_BB2$expression),
  test = NULL,
  na.rm = TRUE,
  parse = TRUE
)

```

```

data_beans_plotBB<- droplevels(subset(data_beans, mkt=="NB"))

stats_results_BB<- ggbetweenstats(data=data_beans_plotBB, x= year, y=gy_kg_ha) %>%
  ↪ extract_stats()
#print(stats_results_BB$pairwise_comparisons_data$p.value)

stats_results_BB3<- stats_results_BB$pairwise_comparisons_data %>%
dplyr::mutate(groups = purrr::pmap(.l = list(group1, group2), .f = c)) %>%
  dplyr::arrange(group1) %>%
  filter(p.value <= 0.05)

nb2<- ggplot(data=data_beans_plotBB, aes(x=reorder(year, -gy_kg_ha), y=gy_kg_ha,
  ↪ fill=year)) +
  ggdist::stat_halfeye(
    adjust = 0.5,
    justification = -0.1,

```

```

    .width = 0,
    point_colour = NA
  ) +

  geom_boxplot(
    width = .12,
    # outlier.color = NA,
    alpha = 0.5)+
  tidyquant::theme_tq()+
  #facet_grid("loc")+
  tidyquant::scale_fill_tq() +
  scale_x_discrete(expand = c(0.03,0)) +
  scale_y_continuous(limits = c(1000,10000), breaks = seq(1000, 10000, by = 1000)) +
  theme(axis.text.x=element_blank(),
        strip.text=element_blank(),
        legend.position = "none",
        panel.grid = element_blank())+
  labs(
    caption=NULL, x=NULL, y=NULL)

nb2<-nb2 +
  ggsignif::geom_signif(
    comparisons = stats_results_BB3$groups,
    map_signif_level = TRUE,
    tip_length = 0.01,
    textsize = 2.7,
    y_position = c(5900, 6300, 6700,7100, 7400, 7800,8200, 8600, 9000, 9400),
    annotations = as.character(stats_results_BB3$expression),
    test = NULL,
    na.rm = TRUE,
    parse = TRUE
  )

```

3.3.1.2 NB

```

data_beans_plotBB<- droplevels(subset(data_beans, mkt=="SR"))

stats_results_BB<- ggbetweenstats(data=data_beans_plotBB, x= loc, y=gy_kg_ha) %>%
  ↪ extract_stats()
#print(stats_results_BB$pairwise_comparisons_data$p.value)

stats_results_BB2<- stats_results_BB$pairwise_comparisons_data %>%
dplyr::mutate(groups = purrr::pmap(.l = list(group1, group2), .f = c)) %>%
  dplyr::arrange(group1)%>%
  filter(p.value <= 0.05)

sr1<- ggplot(data=data_beans_plotBB, aes(x=reorder(loc, -gy_kg_ha), y=gy_kg_ha,
  ↪ fill=loc)) +
  ggdist::stat_halfeye(

```

```

    adjust = 0.5,
    justification = -0.1,
    .width = 0,
    point_colour = NA
  ) +

  geom_boxplot(
    width = .12,
    # outlier.color = NA,
    alpha = 0.5)+
  tidyquant::theme_tq()+
  #facet_grid("loc")+
  tidyquant::scale_fill_tq() +
  scale_x_discrete(expand = c(0.03,0)) +
  scale_y_continuous(limits = c(1000,10000), breaks = seq(1000, 10000, by = 1000)) +
  theme(
    strip.text=element_blank(),
    legend.position = "none",
    panel.grid = element_blank()+
  labs(
    caption=NULL, x="Locations", y="SR")

sr1<-sr1 +
  ggsignif::geom_signif(
    comparisons = stats_results_BB2$groups,
    map_signif_level = TRUE,
    tip_length = 0.01,
    textsize = 2.7,
    y_position = c(5900, 6700, 7500,8300, 9100),
    annotations = as.character(stats_results_BB2$expression),
    test = NULL,
    na.rm = TRUE,
    parse = TRUE
  )

```

```

data_beans_plotBB<- droplevels(subset(data_beans, mkt=="SR"))

stats_results_BB<- ggbetweenstats(data=data_beans_plotBB, x= year, y=gy_kg_ha) %>%
  ↪ extract_stats()
#print(stats_results_BB$pairwise_comparisons_data$p.value)

stats_results_BB3<- stats_results_BB$pairwise_comparisons_data %>%
dplyr::mutate(groups = purrr::pmap(.l = list(group1, group2), .f = c)) %>%
  dplyr::arrange(group1) %>%
  filter(p.value <= 0.05)

sr2<- ggplot(data=data_beans_plotBB, aes(x=reorder(year, -gy_kg_ha), y=gy_kg_ha,
  ↪ fill=year)) +
  ggdist::stat_halfeye(
    adjust = 0.5,
    justification = -0.1,
    .width = 0,

```

```

    point_colour = NA
  ) +

  geom_boxplot(
    width = .12,
    # outlier.color = NA,
    alpha = 0.5)+
  tidyquant::theme_tq()+
  #facet_grid("loc")+
  tidyquant::scale_fill_tq() +
  scale_x_discrete(expand = c(0.03,0)) +
  scale_y_continuous(limits = c(1000,10000), breaks = seq(1000, 10000, by = 1000)) +
  theme(
    strip.text=element_blank(),
    legend.position = "none",
    panel.grid = element_blank()+
  labs(
    caption=NULL, y=NULL, x="Years")

sr2<- sr2 +
  ggsignif::geom_signif(
    comparisons = stats_results_BB3$groups,
    map_signif_level = TRUE,
    tip_length = 0.01,
    textsize = 2.7,
    y_position = c(5900, 6500, 7200,8000, 8700, 9500),
    annotations = as.character(stats_results_BB3$expression),
    test = NULL,
    na.rm = TRUE,
    parse = TRUE
  )

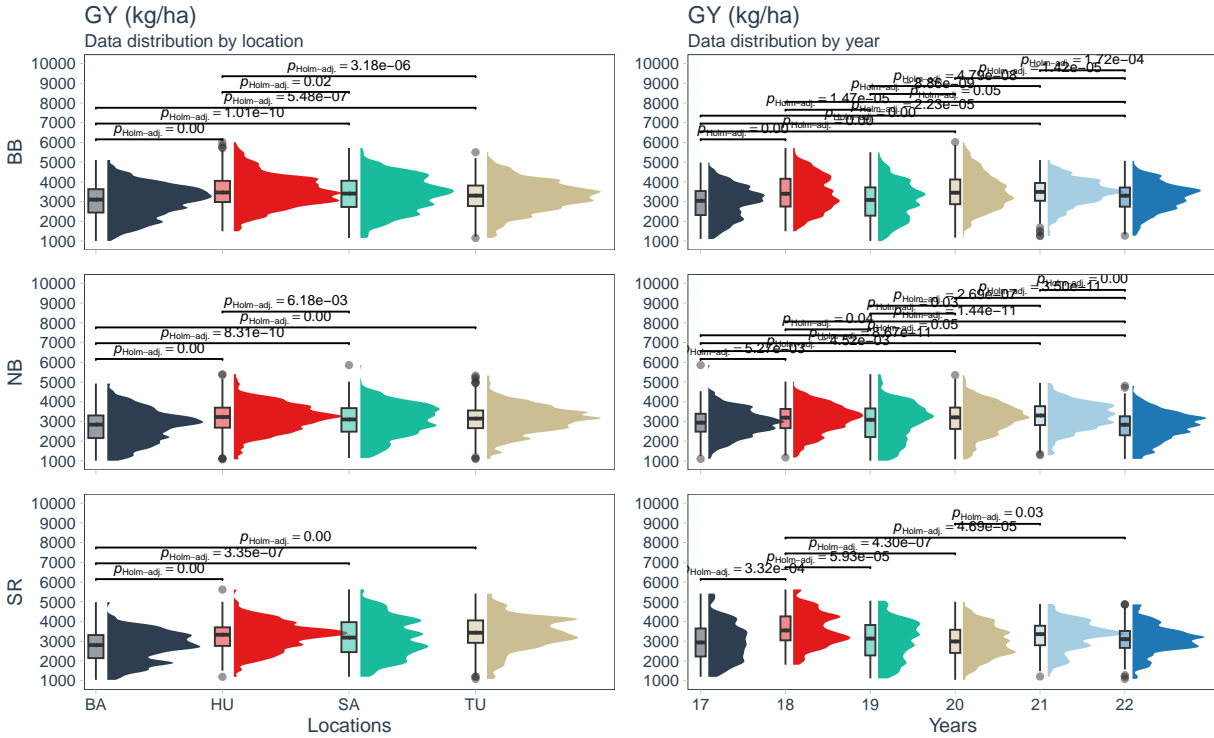
```

3.3.1.3 SR Box plot distribution for each market class in this study across years and locations.

```

print(arrange_ggplot(bb1,bb2,
                     nb1,nb2,
                     sr1,sr2,
                     nrow = 3,
                     ncol = 2))

```



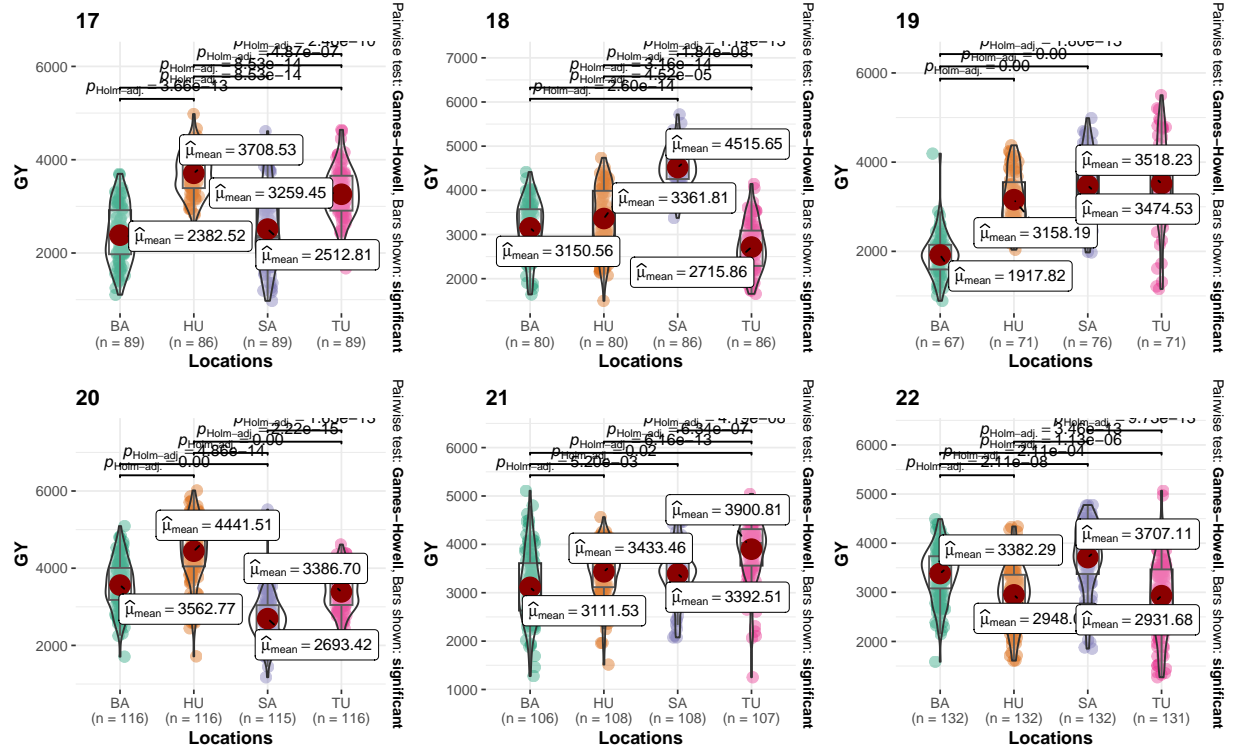
3.3.2 GEI Comparisons (loc:year) - BB

Box plots for between-subjects comparisons by locations using the R package `ggstatsplot`.

- Black beans (BB) **Pairwise Games-Howell test used. Comparisons showing only significant**

```
data_beans_plotBB<- droplevels(subset(data_beans, mkt=="BB"))

plotDM_stats_mkt_loc_BB<- grouped_ggbetweenstats(data=data_beans_plotBB, x= loc,
  ↪ y=gy_kg_ha, type = "parametric", bf.message = F, results.subtitle = F,
  ylab= "GY", xlab = "Locations",
  plot.type = "boxviolin", grouping.var = year )
print(plotDM_stats_mkt_loc_BB)
```

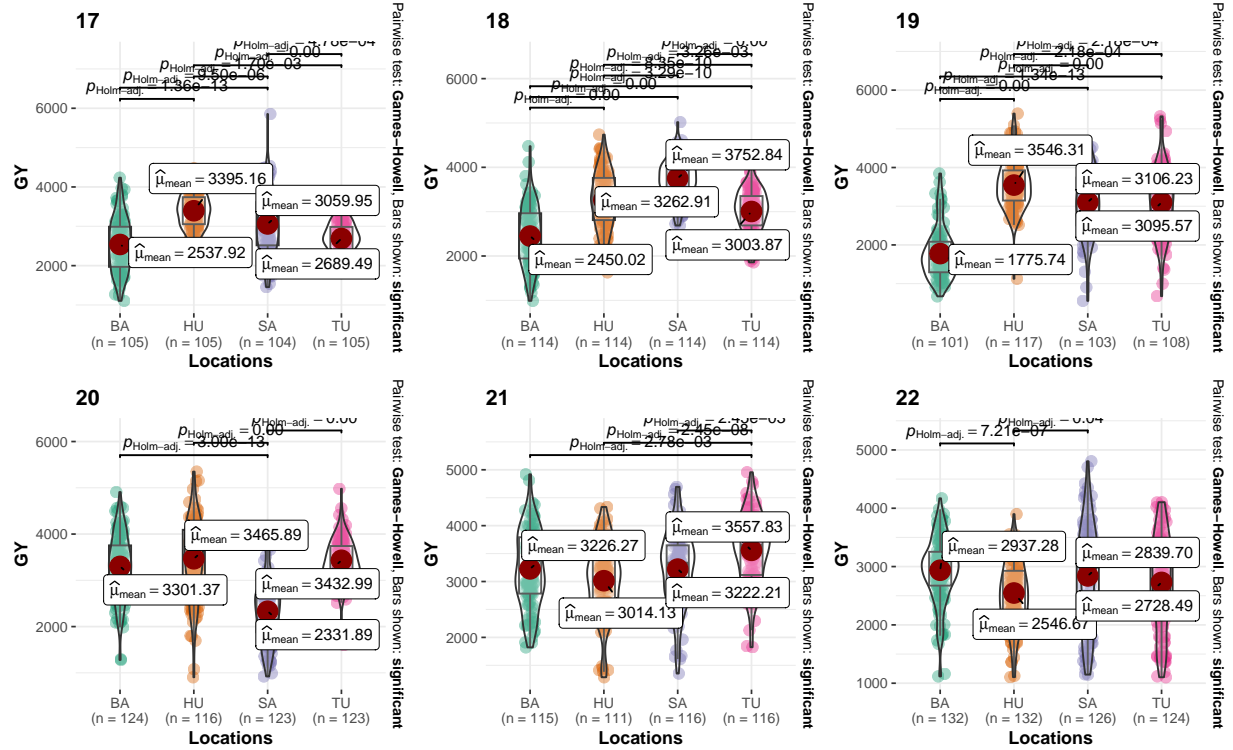
3.3.3 GEI Comparisons (loc:year) - NB

Box plots for between-subjects comparisons by locations using the R package `ggstatsplot`. - Navy beans (NB) **Pairwise Games-Howell test used. Comparisons showing only significant**

```
data_beans_plotNB<- droplevels(subset(data_beans, mkt=="NB"))

plotDM_stats_mkt_loc_NB<- grouped_ggbetweenstats(data=data_beans_plotNB, x= loc,
  ↪ y=gy_kg_ha, type = "parametric", bf.message = F, results.subtitle = F,
  ylab= "GY", xlab= "Locations",
  plot.type = "boxviolin", grouping.var = year )

print(plotDM_stats_mkt_loc_NB)
```



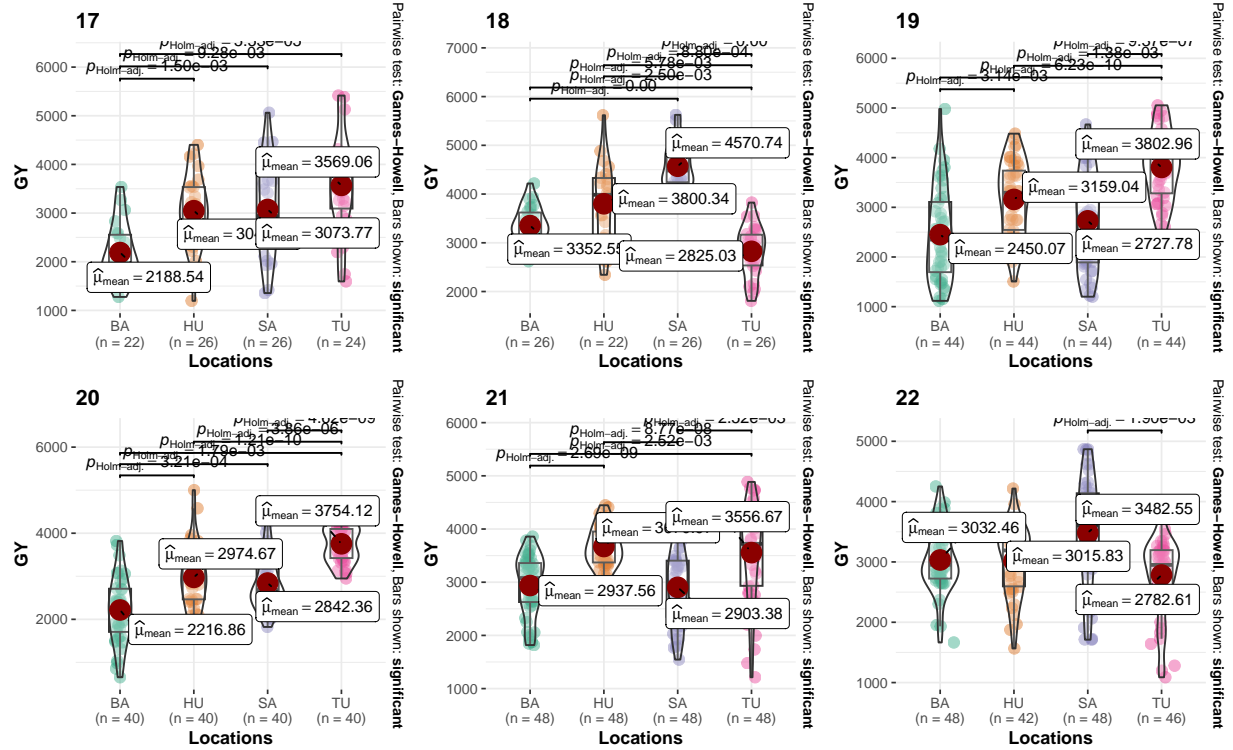
3.3.4 GEI Comparisons (loc:year) - SR

Box plots for between-subjects comparisons by locations using the R package `ggstatsplot`. - Black beans (SR) **Pairwise Games-Howell test used. Comparisons showing only significant**

```
data_beans_plotSR<- droplevels(subset(data_beans, mkt=="SR"))

plotDM_stats_mkt_loc_SR<- grouped_ggbetweenstats(data=data_beans_plotSR, x= loc,
  ↪ y=gy_kg_ha, type = "parametric", bf.message = F, results.subtitle = F,
  ylab= "GY", xlab = "Locations",
  plot.type = "boxviolin", grouping.var = year )

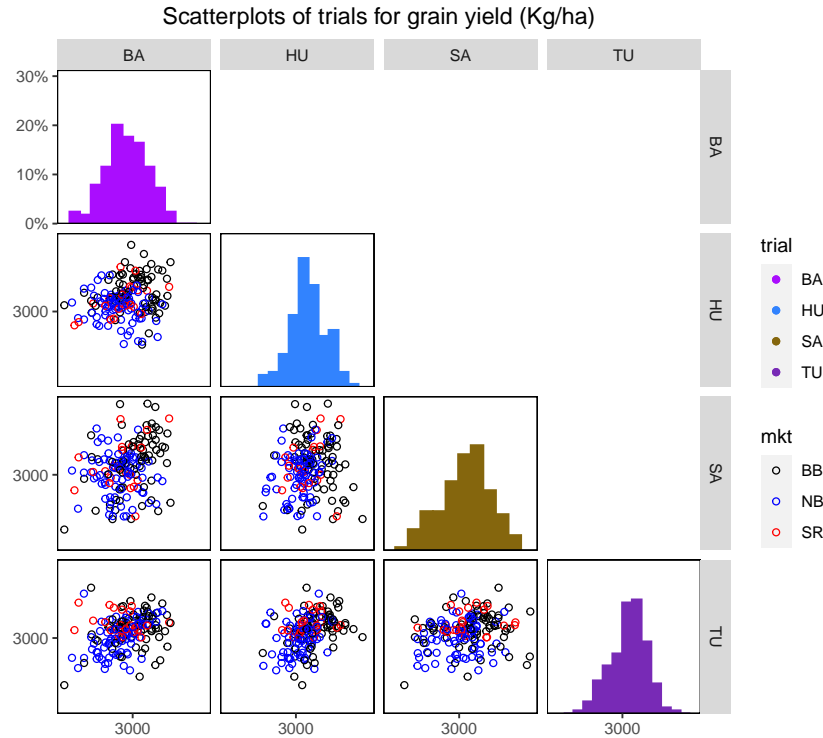
print(plotDM_stats_mkt_loc_SR)
```



3.3.5 Scatter plot

For further insight into the correlation structure between trials a scatter plot matrix can be made using the package `statgenGxE`

```
dropsTD <- statgenSTA::createTD(data = data_beans, genotype = "name", trial = "loc")
options("statgen.genoColors" = c("black", "blue", "red"))
plot(dropsTD, plotType = "scatter", traits = "gy_kg_ha", colorGenoBy = "mkt",
     colorTrialBy = "trial", title = "Scatterplots of trials for grain yield (Kg/ha) ")
```



3.3.6 Outliers

Check for common errors in multi-environment trial by market class data using the R package `metan`. However, the data set was cleaned before at a previous analysis not shown in this vignette using the criteria for GY:

- yield ≤ 0.5 | yield ≥ 7.5 (considering yield as Lb per plot)
- moisture ≥ 30 | moisture ≤ 5 (considering moisture in %)

At this point yield data will not be removed anymore, but we can check it out for further investigation, if it was needed.

```
out_beans<- find_outliers(data_beans, by=mkt, var = gy_kg_ha, plots = F)
```

Trait: gy_kg_ha Number of possible outliers: 9 Line(s): 1113 2052 4478 4522 4539 4620 4739 4740 4761
Proportion: 0.3% Mean of the outliers: 3320 Maximum of the outliers: 5854 | Line 4539 Minimum of the outliers: 554.8 | Line 4739 With outliers: mean = 3015 | CV = 26.27% Without outliers: mean = 3014 | CV = 25.93%

Trait: gy_kg_ha Number of possible outliers: 10 Line(s): 127 1910 3723 3993 4161 4358 5630 6302 6638 6710
Proportion: 0.4% Mean of the outliers: 4313 Maximum of the outliers: 6014 | Line 6302 Minimum of the outliers: 889.9 | Line 3993 With outliers: mean = 3298 | CV = 25.57% Without outliers: mean = 3294 | CV = 25.21%

Trait: gy_kg_ha Number of possible outliers: 4 Line(s): 533 1422 1951 1975 Proportion: 0.4% Mean of the outliers: 4355 Maximum of the outliers: 5625 | Line 1951 Minimum of the outliers: 654.1 | Line 533 With outliers: mean = 3139 | CV = 27.68% Without outliers: mean = 3134 | CV = 27.29%

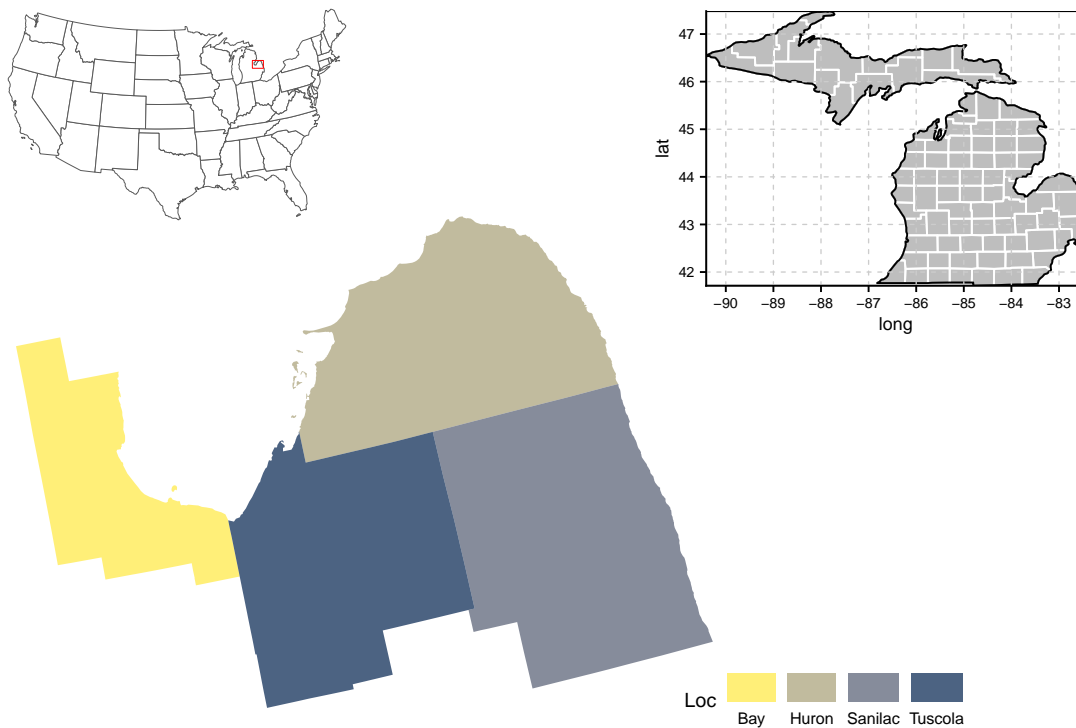
```
#out_beans
```

- No outliers removed after the data inspection based on the Breeder expertise.

3.3.7 Map of locations

This figure represent all counties locations evaluated in this study at Michigan, midwest in the USA.

```
source("utils/map.R")
print(MI_trials_beans_map)
```



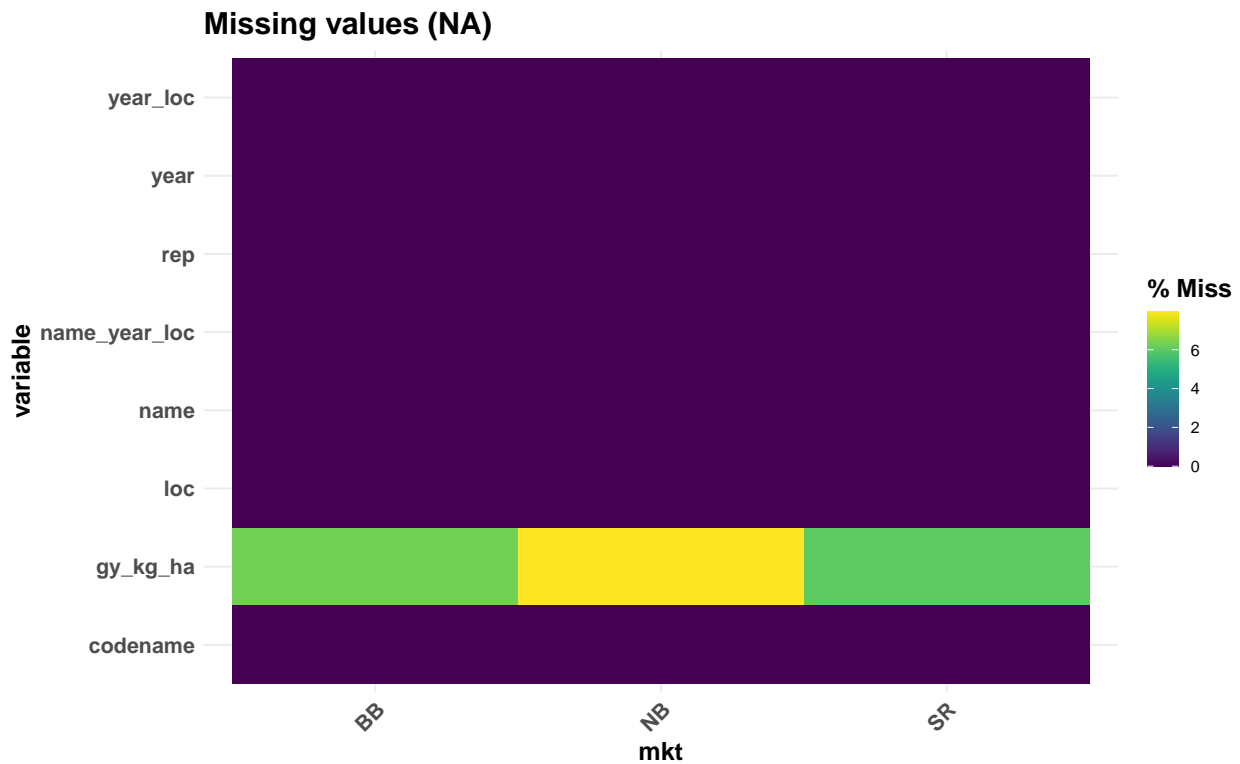
3.3.8 Missing values

```
data_beans.miss.f<- data_beans %>%
  unite(name_year_loc, c(name, year_loc), remove = F) %>%
  group_by(name_year_loc) %>%
  dplyr::summarise(Mean = mean(gy_kg_ha, na.rm = TRUE)) %>%
  filter(!is.na(Mean))

data_beans.miss<- data_beans %>%
  unite(name_year_loc, c(name, year_loc), remove = F)
```

```
data_beans.miss<- data_beans.miss%>%
  filter((name_year_loc %in% data_beans.miss.f$name_year_loc))

gg_miss_fct(data_beans.miss,fct = mkt)+ggtitle("Missing values (NA)") +
  theme(axis.text.x=element_text(face="bold", size = 12),
        axis.text.y = element_text(face = "bold",size = 12) ,
        title = element_text(face = "bold",size = 14))
```



Missing genotypes by market classes

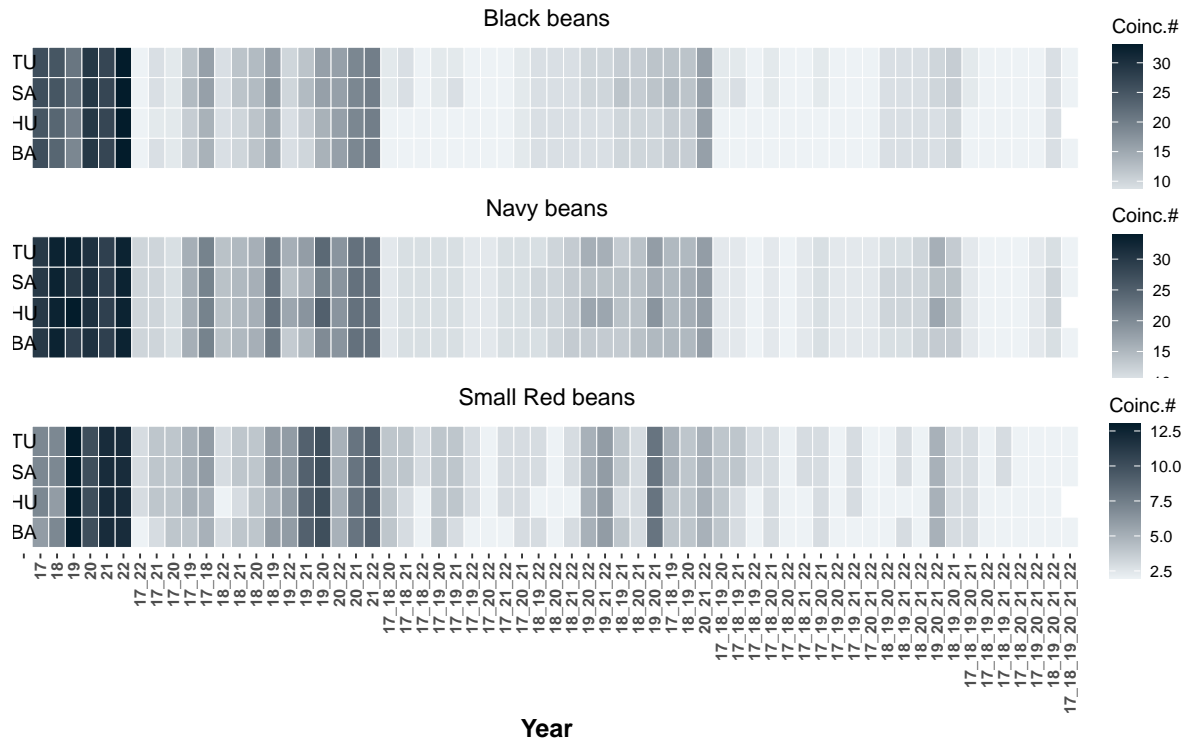
3.3.9 Coincidence Genotypes

Genotypes coincidentes across years combinations

```
source("utils/Coinc.R")
```

```
#> [1] "The coincidence file is present"
```

```
arrange_ggplot(plotB, plotN, plotSR,ncol = 1)
```



Coincidence genotypes per year and combinations of years across locations for each market class.

3.4 Predicted by year mixed model analysis

To predict the BLUPs per year and plot it versus year: BLUPs prediction for the vector of the variable GY in the *i*th genotype, and *j*th loc within year.

3.5 Data preparation

```
#setwd("G:/Shared
↳ drives/Bean_Lab/Volpato/GxE_Variety_trials_Scott/Manuscript/Suppl_mat/TemplateResults-main")
data_beans = read.csv("data/DataBean_MET_GYv2.csv",h=T, stringsAsFactors = T)

# Data adjustment
# All the effect columns must be as a factor to run in ASReml-r.
cols <- c("rep", "name", "loc","year", "mkt", "year_loc")
data_beans[cols] <- lapply(data_beans[cols], factor)
data_beans <- data.table(data_beans)
```

3.5.1 By year mixed model analysis

The BLUPS will be estimated using a mixed-effect model and these values obtained by year using a loop with ASReml and storage into the list.

The following mixed model was used to estimate the BLUPs of each genotype within year with one value per genotype per year (random effects are underlined in all equations):

$$Y_{ijl} = \mu + \underline{G_i} + E_l + \beta_{jl} + \underline{GE_{il}} + \underline{\varepsilon_{ijl}}$$

where Y_{ijl} is the response variable (e.g., grain yield) observed in the j th repetition of the i th genotype in the l th location ($i = 1, 2, \dots, g$; $j = 1, 2, \dots, b$; $l = 1, 2, \dots, e$); μ is the grand mean; $\underline{G_i}$ is the effect of the i th genotype; E_l is the effect of the l th location (env); β_{jl} is the effect of the j th rep with the l th location; $\underline{GE_{il}}$ is the interaction effect of the i th genotype nested within the l th location; and $\underline{\varepsilon_{ijl}}$ is the random error, in witch with $G_i \sim N(0, \sigma_G^2)$, $GE_{il} \sim N(0, \sigma_{GE}^2)$, and $\varepsilon_{ijl} \sim N(0, \sigma_\varepsilon^2)$, all independent, where G_G^2 is the genotype (name) variance, GS_{GE}^2 is the interaction genotype x environment variance, and σ_ε^2 is the mean error variance across experiments.

```
## Analysis per site and mkt class
mkt_n <- levels(data_beans$mkt)

Envs <- levels(data_beans$year)

stgI_list <- matrix(data=list(), nrow=length(Envs), ncol=1,
                    dimnames=list(Envs, c("BLUPS")))

mkt <- nlevels(data_beans$mkt)

for(k in 1:mkt){

  bk <- levels(data_beans$mkt)
  cj <- bk[k]
  #print(cj)

  data_beans_temp <- droplevels(subset(data_beans, mkt==cj))

  for (i in Envs){
    #i=Envs[1]
    Edat <- droplevels(subset(data_beans_temp, year==i))

    #print(i)

    mod.1 <- asreml(fixed      = gy_kg_ha ~ loc + loc:rep,
                   random     = ~ name + name:loc,
                   data       = Edat,
                   predict    = predict.asreml(classify = "name"),
                   trace      = F,
                   maxit      = 500)

    # print(summary.asreml(mod.1)$varcomp)
    # wald(mod.1)

    blup.1 <- data.table((mod.1$predictions$pvals[1:3]))
    names(blup.1) <- c("name", "yield", "se")
    blup.1$mkt <- cj

    stgI_list[[i, "BLUPS"]] <- blup.1 # put all the results of Stage 1 in the list

    # rm(Edat, mod.1, blue, blup.1)
```



```

}
  if(k==1){stgI_list.1<-stgI_list}else{stgI_list.1<-rbind(stgI_list.1, stgI_list)}
}

```

3.5.1.1 Preparing dataset Merging the original data to have all the factors in the final table with: name, year, and mkt

```

##### Unlist the results of Stage I and format as data.table #####
blups_stageI <- data.table(ldply(stgI_list.1[, "BLUPS"], data.frame, .id="year"))

blues_stage.I <- blups_stageI[order(blups_stageI$year,blups_stageI$mkt),]

# Change the order of columns
blues_stage.I <- blues_stage.I %>%
  dplyr::select(year, name, mkt, se, yield)

str(blues_stage.I)

```

Classes 'data.table' and 'data.frame': 984 obs. of 5 variables: \$ year : Factor w/ 6 levels "17","18","19",...: 1 1 1 1 1 1 1 1 1 1 ... \$ name : Factor w/ 164 levels "B1","B10","B11",...: 1 2 3 4 5 6 7 8 9 10 ... \$ mkt : chr "BB" "BB" "BB" "BB" ... \$ se : num 141.3 98.7 113.8 113.8 355.7 ... \$ yield: num 2727 3579 2947 3092 2959 ... - attr(*, "internal.selfref")=

3.5.2 Descriptive variance

```

if (knitr::is_html_output()) {

  print_table(blues_stage.I)

}else{

flextable(head(blues_stage.I)) %>%
  add_footer_lines(
    c("Beans data set by year",
      "Header data set showing the 6 first entry")) %>%
  autofit() %>%
  add_header_lines("Dry Beans varieties trial") %>%
  theme_design2()

}

```

3.5.2.1 Beans data set by year

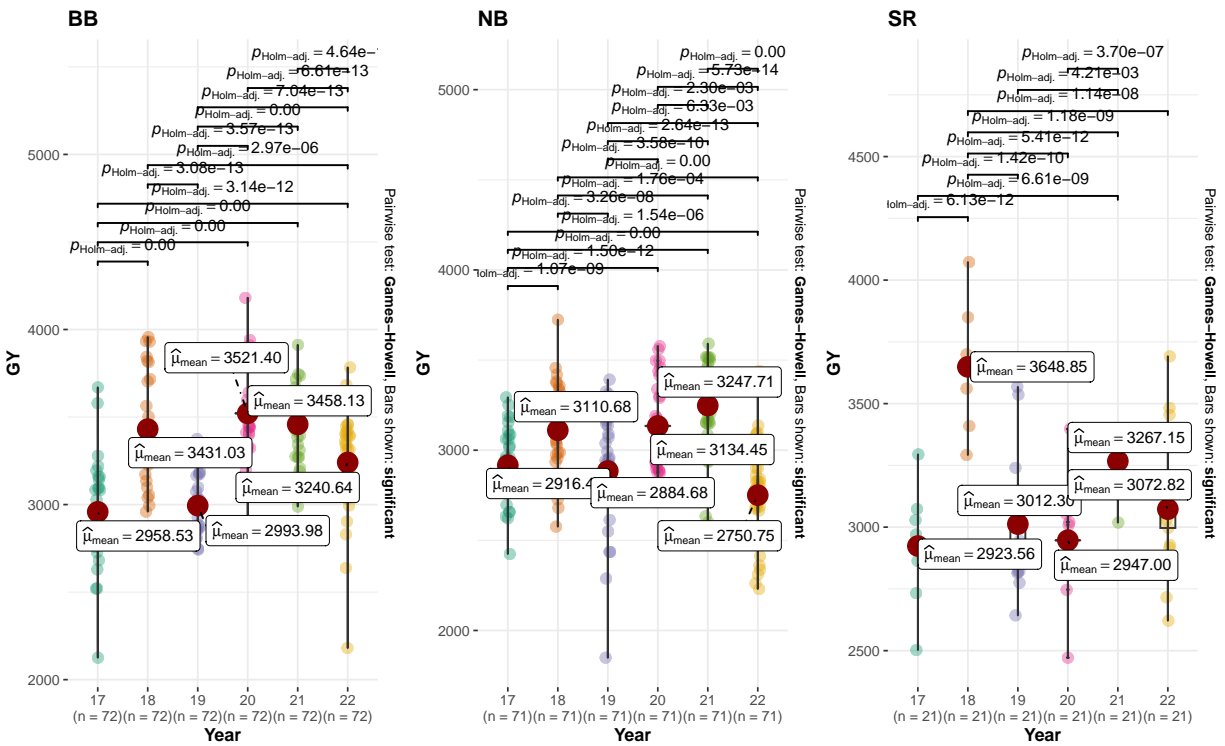
Dry Beans varieties trial

year	name	mkt	se	yield
17	B1	BB	141.29	2,727
17	B10	BB	98.73	3,579
17	B11	BB	113.83	2,947
17	B12	BB	113.83	3,092
17	B13	BB	355.72	2,959
17	B14	BB	355.72	2,959

Beans data set by year

Header data set showing the 6 first entry

3.5.2.2 Box plot distribution Pairwise Games-Howell test used. Comparisons showing only significant



Combination of box and violin plots along with jittered data points for between subjects comparisons by years of grain yield (GY) for black (BB), Navy (NB) and Red (SR) beans. Pairwise Games-Howell test used. Comparisons showing only significant values.

```
str(blues_stage.I)
```

Classes 'data.table' and 'data.frame': 984 obs. of 5 variables: \$ year : Factor w/ 6 levels "17","18","19",...: 1 1 1 1 1 1 1 1 1 1 ... \$ name : Factor w/ 164 levels "B1","B10","B11",...: 1 2 3 4 5 6 7 8 9 10 ... \$ mkt : chr "BB" "BB" "BB" "BB" ... \$ se : num 141.3 98.7 113.8 113.8 355.7 ... \$ yield: num 2727 3579 2947 3092 2959 ... - attr(*, "internal.selfref")=

```
blues_stage.I$year <- as.numeric(blues_stage.I$year)
```

```
blues_stage.I$year[blues_stage.I$year == 1]<- 17
```

```
blues_stage.I$year[blues_stage.I$year == 2]<- 18
```

```
blues_stage.I$year[blues_stage.I$year == 3]<- 19
```

```
blues_stage.I$year[blues_stage.I$year == 4]<- 20
```

```
blues_stage.I$year[blues_stage.I$year == 5]<- 21
```

```
blues_stage.I$year[blues_stage.I$year == 6]<- 22
```

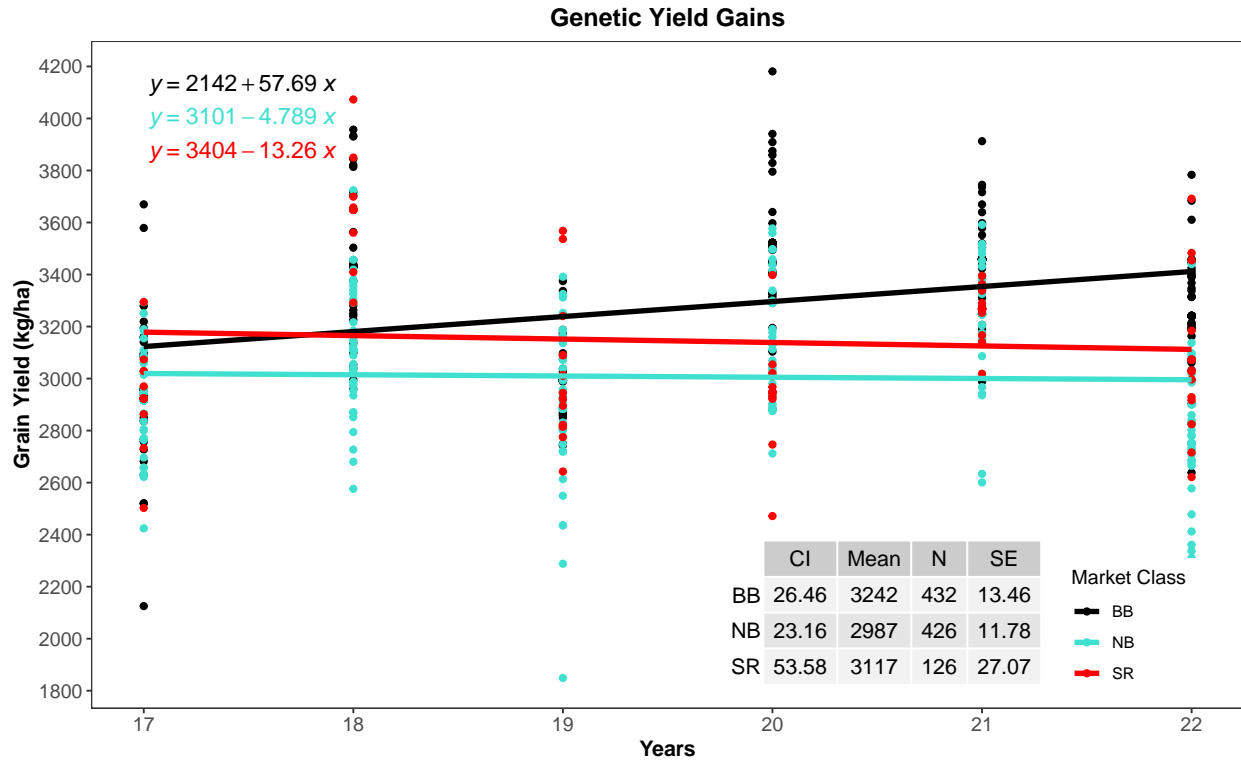
```
beans_desc <- desc_stat(blues_stage.I, yield, stats = ("ci.t, hmean, n, se"), by = mkt)
#print(beans_desc)
```

```
beans_desc<-beans_desc[-c(2)]
```

```
colnames(beans_desc)<- c("MKT","CI", "Mean", "N", "SE")
```

```
beans_desc<-data.frame(beans_desc, row.names =T)
```

```
#rownames(beans_desc) <- NULL
```



Combination of box and violin plots along with jittered data points for between subjects comparisons by years of grain yield (GY) for black (BB), Navy (NB) and Red (SR) beans. Pairwise Games-Howell test used. Comparisons showing only significant values.

3.6 Two Stage mixed model analysis

The following pipeline will be used: 1-Stage = BLUEs estimation for the vector of the variable GY in the *i*th genotype, and *j*th year within *loc*. Then the BLUEs from the 1-Stage (Y_{ik}) will be used to predict the BLUPs (Y_{ijl}) of the *i*th genotype in the *l*th location and *j*th rep in the 2-Stage in which this second model have *name* and *loc* effects as random.

3.6.1 1st Stage mixed model analysis

```
#rm(list=ls())
data_beans = read.csv("data/DataBean_MET_GYv2.csv",h=T, stringsAsFactors = T)
#str(data_beans)
# Data adjustment
# All the effect columns must be as a factor to run in ASReml-r.
cols <- c("rep", "name", "loc","year", "mkt", "year_loc")
data_beans[cols] <- lapply(data_beans[cols], factor)
data_beans <- data.table(data_beans)
```

The BLUEs will be estimated using a mixed-effect model. The BLUEs will be obtained by location using a loop with ASReml and storage into the list.

The following mixed model was used to estimate the BLUEs of each genotype within location with one value per genotype per experiment and rep, for the first step (random effects are underlined in all equations):

$$Y_{ik} = \mu + G_i + S_k + GS_{ik} + \varepsilon_{ik}$$

where Y_{ik} is the observed yield in the i th genotype and k th year, μ is the overall mean, G_i is the effect of the i th genotype, S_k is the effect of the k th year, GS_{ik} is the effect of the interaction between the i th genotype and the k th year, and ε_{ik} are the residual, with $S_k \sim N(0, \sigma_Y^2)$, $GS_{ik} \sim N(0, \sigma_Y^2)$, and $\varepsilon_{ik} \sim N(0, \sigma_\varepsilon^2)$, all independent, where σ_Y^2 is the year variance, and σ_ε^2 is the mean error variance across experiments.

```
#> `summarise()` has grouped output by 'name'. You can override using the `.groups` argument.
#> `summarise()` has grouped output by 'name'. You can override using the `.groups` argument.
#> `summarise()` has grouped output by 'name'. You can override using the `.groups` argument.
#> `summarise()` has grouped output by 'name'. You can override using the `.groups` argument.
#> `summarise()` has grouped output by 'name'. You can override using the `.groups` argument.
#> `summarise()` has grouped output by 'name'. You can override using the `.groups` argument.
#> `summarise()` has grouped output by 'name'. You can override using the `.groups` argument.
#> `summarise()` has grouped output by 'name'. You can override using the `.groups` argument.
#> `summarise()` has grouped output by 'name'. You can override using the `.groups` argument.
#> `summarise()` has grouped output by 'name'. You can override using the `.groups` argument.
```

```
## Loop to get the year effect corrected by year
## Only genotypes present in years

#str(data_beans)
data_beans$var_years <- as.factor(data_beans$var_years)
year <- nlevels(data_beans$var_years)
## Analysis per site and mkt class
#mkt_n <- levels(data_beans$mkt)

Envs <- levels(data_beans$loc)

stgI_list <- matrix(data=list(), nrow=length(Envs), ncol=1,
                    dimnames=list(Envs, c("BLUES")))

mkt <- nlevels(data_beans$mkt)

for (y in 1:year) {

  by <- levels(data_beans$var_years)
  cy <- by[y]

  data_beans_temp1 <- droplevels(subset(data_beans, var_years==cy))

  if(y == 1) {next
} else {

  for(k in 1:mkt){

    bk <- levels(data_beans_temp1$mkt)
    cj <- bk[k]
    #print(cj)
```

```

data_beans_temp2 <- droplevels(subset(data_beans_temp1, mkt==cj))

for (i in Envs){
  #i=Envs[1]
  Edat <- droplevels(subset(data_beans_temp2, loc==i))

  #print(i)

  mod.1 <- asreml(fixed      = gy_kg_ha ~ name:rep,
                  random    = ~ year + name:year ,
                  data       = Edat,
                  predict    = predict.asreml(classify = "name:rep",vcov=TRUE, aliased =
                    ↪ T),
                  trace      = F,
                  maxit      = 500)

  # print(summary.asreml(mod.1)$varcomp)
  # wald(mod.1)

  blue.1<- data.table((mod.1$predictions$pvals[1:4]))
  names(blue.1) <- c("name", "rep", "yield", "se")
  blue.1$mkt<- cj
  blue.1$var_years<- cy

  stgI_list[[i, "BLUES"]] <- blue.1 # put all the results of Stage 1 in the list

  rm(Edat,mod.1, blue, blue.1)

}
if(k==1){stgI_list.1<-stgI_list}else{stgI_list.1<-rbind(stgI_list.1, stgI_list)}
}
}
if(y==2){stgI_list.2<-stgI_list.1}else{stgI_list.2<-rbind(stgI_list.2, stgI_list.1)}
}

```

3.6.1.1 Preparing dataset of Stage I for Stage II Merging the original data to have all the factors in the final table with: name, loc, mkt, rep

```

##### Unlist the results of Stage I and format as data.table #####
blues_stageI <- data.table(lapply(stgI_list.2[, "BLUES"], data.frame, .id="loc"))

blues_stage.I <- blues_stageI[order(blues_stageI$loc,blues_stageI$name),]

blues_stage.I.SEmean <- blues_stage.I %>%
  dplyr::summarise(Mean_SE = mean(se, na.rm = TRUE))

print(blues_stage.I.SEmean)

```

Mean_SE 1 433.5

```
#str(blues_stage.I)
```

```
# Change the order of columns
blues_stage.I <- blues_stage.I %>%
  dplyr::select(loc, name, rep, mkt, yield)
```

3.6.2 2nd Stage mixed model analysis

The following linear mixed model with interaction effect will be used in the 2-Stage in order to investigate the multi-environment trials (MET) as follow:

$$Y_{ijl} = \mu + G_i + E_l + \beta_{jl} + GE_{il} + \varepsilon_{ijl}$$

where Y_{ijl} is the response variable (e.g., grain yield) observed in the j th repetition of the i th genotype in the l th location ($i = 1, 2, \dots, g$; $j = 1, 2, \dots, b$; $l = 1, 2, \dots, e$); μ is the grand mean; G_i is the effect of the i th genotype; E_l is the effect of the l th location (env); β_{jl} is the effect of the j th rep with the l th location; GE_{il} is the interaction effect of the i th genotype nested within the l th location; and ε_{ijl} is the random error, in witch with $G_i \sim N(0, \sigma_G^2)$, $GE_{il} \sim N(0, \sigma_{GE}^2)$, and $\varepsilon_{ijl} \sim N(0, \sigma_\varepsilon^2)$, all independent, where G_G^2 is the genotype (name) variance, GS_{GE}^2 is the interaction genotype x environment variance, and σ_ε^2 is the mean error variance across experiments.

3.6.3 By market classes

- Getting the files for the individually market classes

```
blues_stage.I_BB <- droplevels(subset(blues_stage.I, mkt=="BB"))
blues_stage.I_NB <- droplevels(subset(blues_stage.I, mkt=="NB"))
blues_stage.I_SR <- droplevels(subset(blues_stage.I, mkt=="SR"))
```

```
blues_stage.I_table1 <- blues_stage.I %>%
  group_by(loc, rep, mkt) %>%
  dplyr::summarise(count = length(name))

blues_stage.I_table1 <- blues_stage.I_table1[, 3:4]
blues_stage.I_table1 <- blues_stage.I_table1[1:3,]

blues_stage.I_table1
```

mkt	count
factor	integer
BB	72
NB	71
SR	21

```
str(blues_stage.I_BB)
```

Classes 'data.table' and 'data.frame': 1152 obs. of 5 variables: \$ loc : Factor w/ 4 levels "BA","HU","SA",...: 1 1 1 1 1 1 1 1 1 1 ... \$ name : Factor w/ 72 levels "B13","B14","B19",...: 1 1 1 1 2 2 2 2 3 3 ... \$ rep : Factor w/ 4 levels "1","2","3","4": 1 2 3 4 1 2 3 4 1 2 ... \$ mkt : Factor w/ 1 level "BB": 1 1 1 1 1 1 1 1 1 1 ... \$ yield: num 2626 2874 2391 3000 2851 ... - attr(*, ".internal.selfref")=

```
if (knitr::is_html_output()) {
  print_table(blues_stage.I_BB)
}else{
  flextable(head(blues_stage.I_BB)) %>%
    add_footer_lines(
      c("Black beans data set",
        "Header data set showing the 6 first entry")) %>%
    autofit() %>%
    add_header_lines("Dry Beans varieties trial") %>%
    theme_design2()
}
```

Dry Beans varieties trial				
loc	name	rep	mkt	yield
BA	B13	1	BB	2,626
BA	B13	2	BB	2,874
BA	B13	3	BB	2,391
BA	B13	4	BB	3,000
BA	B14	1	BB	2,851
BA	B14	2	BB	3,235
Black beans data set				
Header data set showing the 6 first entry				

```
if (knitr::is_html_output()) {
  print_table(blues_stage.I_NB)
}else{
  flextable(head(blues_stage.I_NB)) %>%
    add_footer_lines(
      c("Navy beans data set",
```



```

      "Header data set showing the 6 first entry")) %>%
autofit() %>%
add_header_lines("Dry Beans varieties trial") %>%
theme_design2()
}

```

Dry Beans varieties trial				
loc	name	rep	mkt	yield
BA	N12	1	NB	2,505
BA	N12	2	NB	2,642
BA	N12	3	NB	2,862
BA	N12	4	NB	1,323
BA	N13	1	NB	2,451
BA	N13	2	NB	2,673
Navy beans data set				
Header data set showing the 6 first entry				

```

if (knitr::is_html_output()) {

  print_table(blues_stage.I_SR)

}else{

flextable(head(blues_stage.I_SR)) %>%
  add_footer_lines(
    c("Red beans data set",
      "Header data set showing the 6 first entry")) %>%
  autofit() %>%
  add_header_lines("Dry Beans varieties trial") %>%
  theme_design2()
}

```

Dry Beans varieties trial				
loc	name	rep	mkt	yield
BA	R10	1	SR	2,296
BA	R10	2	SR	2,676
Red beans data set				
Header data set showing the 6 first entry				

Dry Beans varieties trial

loc	name	rep	mkt	yield
BA	R10	3	SR	3,440
BA	R10	4	SR	2,580
BA	R12	1	SR	3,001
BA	R12	2	SR	2,716

Red beans data set

Header data set showing the 6 first entry

3.6.4 Descriptive variance**3.6.4.1 Box plot distribution**

- Box plots for between-subjects comparisons by locations using the R package `ggstatsplot`.

Pairwise Games-Howell test used. Comparisons showing only significant

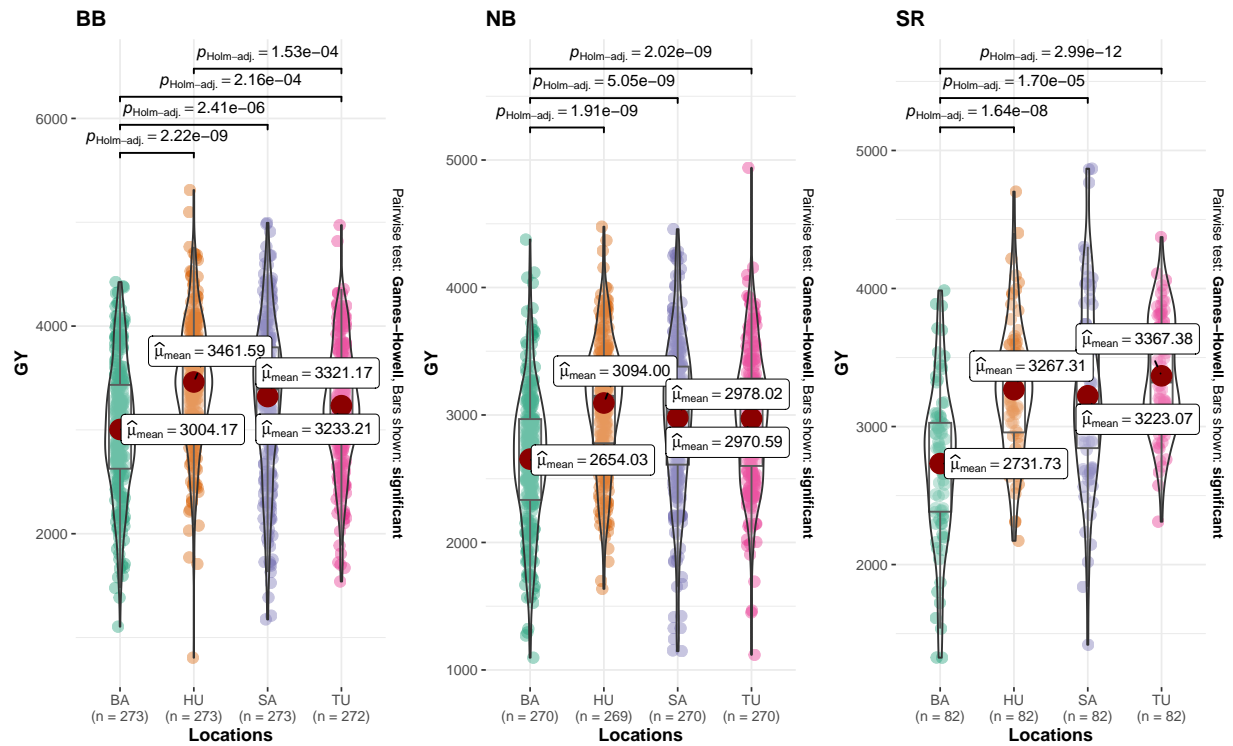


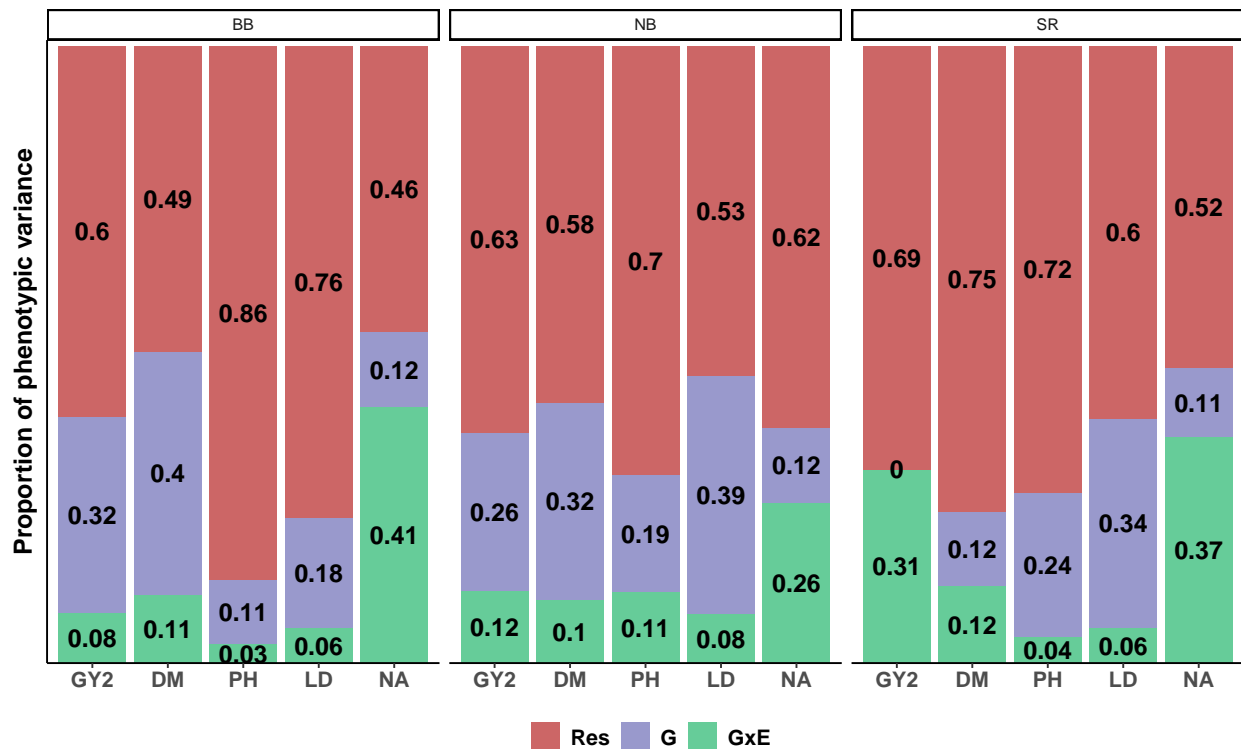
Figure S4: Combination of box and violin plots along with jittered data points for between subjects comparisons by locations of grain yield (GY) for black (BB), Navy (NB) and Red (SR) beans. Pairwise Games-Howell test used. Comparisons showing only significant values. BA: Bay, HU: Huron, SA:Sanilac, TU: Tuscola.

```

data_beans_var = read.csv("data/VarStack.csv",h=T, stringsAsFactors = T)
data_beans_var$trait <- factor(data_beans_var$trait, levels=c("GY1", "GY2", "DM", "PH",
  ↪ "LD"))

ggplot(data_beans_var, aes(x = trait, y = Freq, fill = Comp, label = Comp)) +
  geom_bar(stat = "identity") +
  #geom_text(size = 3, position = position_stack(vjust = 0.5)) +
  facet_wrap("mkt") +
  geom_text(aes(label = round(Freq, 2), x = trait, y = Freq), data = data_beans_var, size
  ↪ = 5, position = position_stack(vjust = 0.5), fontface = "bold") +
  labs(y = "Proportion of phenotypic variance", fill = "", x = NULL) +
  theme_classic() +
  #theme(strip.background = element_blank()) +
  scale_fill_manual(values=c("#CC6666", "#9999CC", "#66CC99"),
    labels = c("Res", "G", "GxE")) +
  theme(legend.position = "bottom",
    axis.text.x=element_text(face="bold", size = 12),
    axis.text.y=element_blank(),
    axis.ticks.y = element_blank(),
    axis.title.y=element_text(face="bold", size = 14) ,
    legend.text = element_text(size = 12, face="bold"), ) +
  scale_y_continuous(expand = expansion(mult = c(0, 0.01)))

```



Proportion of the phenotypic variance for three dry beans traits evaluated across three locations. GY1, grain yield from 2017 to 2022; GY2, grain yield only in 2021; DM, days to maturity; PH, plant height; LD, lodging.

3.6.4.2 Phenotypic variance components

```

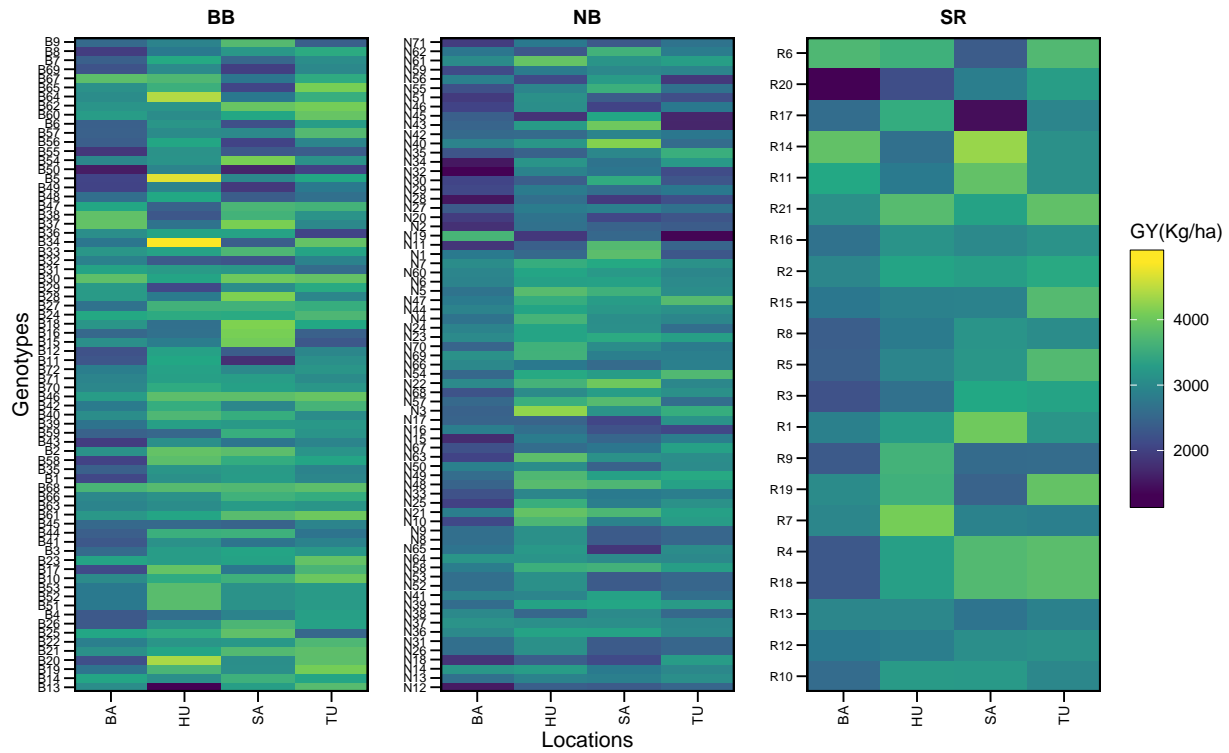
blues_stage.I_BB_ge<- na.omit(blues_stage.I_BB)
stab_bb <- ge_plot(blues_stage.I_BB_ge, type = 1,
  env = loc,
  gen = name,
  resp = yield) + xlab("") + ylab("Genotypes") +
theme(axis.text.x = element_text(size = 8, angle = 90, vjust = 0.5, hjust = 1),
  axis.text.y = element_text(size = 7, angle = 0, vjust = 0.5, hjust = 1),
  legend.position = "none", plot.title=element_text(hjust=0.5)) +
  labs(title = "BB")
#print (stab_bb)

blues_stage.I_NB_ge<- na.omit(blues_stage.I_NB)
stab_nb <- ge_plot(blues_stage.I_NB_ge, type = 1,
  env = loc,
  gen = name,
  resp = yield) + xlab("Locations") + ylab("") +
theme(axis.text.x = element_text(size = 8, angle = 90, vjust = 0.5, hjust = 1),
  axis.text.y = element_text(size = 7, angle = 0, vjust = 0.5, hjust = 1),
  legend.position = "none", plot.title=element_text(hjust=0.5)) +
  labs(title = "NB")
#print (stab_nb)

blues_stage.I_SR_ge<- na.omit(blues_stage.I_SR)
stab_sr <- ge_plot(blues_stage.I_SR_ge, type = 1,
  env = loc,
  gen = name,
  resp = yield) + xlab("") + ylab("") +
theme(axis.text.x = element_text(size = 8, angle = 90, vjust = 0.5, hjust = 1),
  axis.text.y = element_text(size = 7, angle = 0, vjust = 0.5, hjust = 1),
  legend.position = "right",
  plot.title=element_text(hjust=0.5)) +
  labs(fill="GY(Kg/ha)", title = "SR")
#print (stab_sr)

print(arrange_ggplot(stab_bb, stab_nb,stab_sr))

```



Genotype's performance across the environments for Black (BB), Navy (NB), and Small Red (SR) beans using the estimated means (BLUES) from the 1-stage mixed model analysis.

3.6.4.3 Genotyping performance

3.7 Descriptive MET

- Genetic correlations across environment using the Unstructured (US) variance-covariance structure.

This mixed model will be used only for genotype x environment correlations in order to investigate the GEI correlations.

```
mod.us.bb <- asreml(fixed = yield ~ loc + loc:rep ,
                    random  = ~ name:us(loc) ,
                    data     = blues_stage.I_BB,
                    predict  = predict.asreml(classify = "name"),
                    trace    = F,
                    maxit    = 500)

#print(wald(mod.us.bb))
#print(summary.asreml(mod.us.bb)$varcomp)

f=summary(mod.us.bb)$varcomp[1:10,1]
z=matrix(0, 4,4)
z[upper.tri(z)| row(z)==col(z)] <- f
corf=z/sqrt(diag(z)%*%t(diag(z)))
#corf
```

```

rownames(corf)=c(
"BA",
"HU",
"SA",
"JU")

colnames(corf)=rownames(corf)

plotBB<- ggcorrplot(corf, colors = c("#6D9EC1", "white", "#E46726"),
  title = "
      Black Beans",
  show.legend = F,
  legend.title = "rgg" ,lab_size=5,tl.srt = 90,type = c("upper"), lab = T,digits = 4,
  outline.color = "white",pch.col = "white", tl.col = "blue",show.diag = FALSE)
## genetic correlation: manual estimation
# corr.loc_BA_HU<-summary(mod.us.bb)$varcomp[2,1] /
#   sqrt(summary(mod.us.bb)$varcomp[1,1]*summary(mod.us.bb)$varcomp[3,1])

# print(plotBB)

mod.us.nb <- asreml(fixed = yield ~ loc + loc:rep ,
  random      = ~ name:us(loc) ,
  data        = blues_stage.I_NB,
  predict     = predict.asreml(classify = "name"),
  trace       = F,
  maxit       = 500)

#print(wald(mod.us.nb))
#print(summary.asreml(mod.us.nb)$varcomp)

f=summary(mod.us.nb)$varcomp[1:10,1]
z=matrix(0, 4,4)
z[upper.tri(z)| row(z)==col(z)] <- f
corf=z/sqrt(diag(z)%*%t(diag(z)))
#corf

rownames(corf)=c(
"BA",
"HU",
"SA",
"JU")

colnames(corf)=rownames(corf)

#corf<- as_tibble(corf,rownames=NA)

plotNB<-ggcorrplot(corf, colors = c("#6D9EC1", "white", "#E46726"),
  title = "
      Navy Beans",
  show.legend = F,
  legend.title = "r" ,lab_size=5,tl.srt = 90,type = c("upper"), lab = T,digits = 4,
  outline.color = "white",pch.col = "white", tl.col = "blue",show.diag = FALSE)
## genetic correlation: manual estimation
corr.loc<-summary(mod.us.nb)$varcomp[2,1] /

```

```

  sqrt(summary(mod.us.nb)$varcomp[1,1]*summary(mod.us.nb)$varcomp[3,1])
#print(plotNB)

mod.us.sr <- asreml(fixed = yield ~ loc + loc:rep ,
                  random  = ~ name:us(loc) ,
                  data     = blues_stage.I_SR,
                  predict  = predict.asreml(classify = "name"),
                  trace    = F,
                  maxit    = 50000)

#print(wald(mod.us.sr))
#print(summary.asreml(mod.us.sr)$varcomp)

f=summary(mod.us.sr)$varcomp[1:10,1]
z=matrix(0, 4,4)
z[upper.tri(z)| row(z)==col(z)] <- f
corf=z/sqrt(diag(z)%*%t(diag(z)))
#corf

rownames(corf)=c(
"BA",
"HU",
"SA",
"TI")

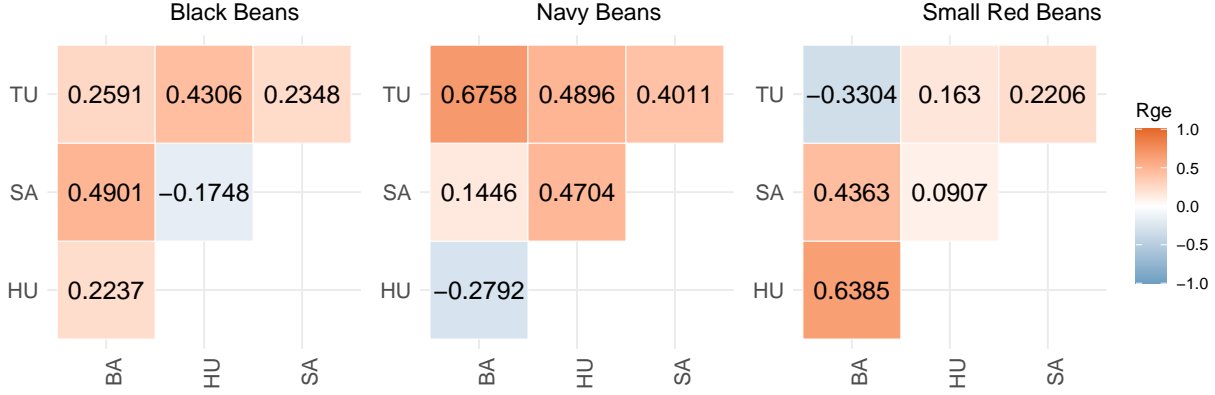
colnames(corf)=rownames(corf)

#corf<- as_tibble(corf,rownames=NA)

plotSR<- ggcorrplot(corf, colors = c("#6D9EC1", "white", "#E46726"),
                  title = "
                        Small Red Beans",
                  #show.legend = F,
                  legend.title = "Rge" ,lab_size=5,tl.srt = 90,type = c("upper"), lab = T,digits = 4,
                  outline.color = "white",pch.col = "white", tl.col = "blue",show.diag = FALSE)
## genetic correlation: manual estimation
corr.loc<-summary(mod.us.sr)$varcomp[2,1] /
  sqrt(summary(mod.us.sr)$varcomp[1,1]*summary(mod.us.sr)$varcomp[3,1])
#print(plotSR)

arrange_ggplot(plotBB,plotNB, plotSR,ncol = 3)

```



3.8 Analyzing MET - GGE

Genotype plus Genotype-vs-Environment interaction (GGE) model has been widely used to genotype evaluation and mega-environment identification in multi-environment trials (MET). This model considers a GGE (i.e., G + GE) biplot, which is constructed by the first two symmetrically scaled principal components (PC1 and PC2) derived from singular value decomposition of environment-centered MET data. The GGE biplot graphically displays G plus GE of a MET in a way that facilitates visual genotype evaluation and mega-environment identification Yan, Kang, Ma, Woods, & Cornelius (2007).

3.8.1 GGE biplot model

The mean yield of genotype i in environment j is commonly described by a general linear model

$$\hat{y}_{ij} + \mu + \alpha_i + \beta_j + \phi_{ij}$$

where \hat{y}_{ij} is the mean yield of genotype i in environment j , $i = 1, \dots, g$; $j = 1, \dots, e$ being g and e the numbers of genotypes and environments, respectively; μ is the grand mean; α_i is the main effect of the genotype i ; β_j is the main effect of the environment j , and ϕ_{ij} is the interaction effect between genotype i and environment j . In the Genotype plus Genotype-vs-Environment interaction (GGE) model the α_i term is deleted from the above model and then the environment-centered data matrix, ϕ_{ij} , is subjected to SVD Yan & Kang (2003) and Yan et al. (2007). Explicitly, we have

$$\phi_{ij} = \hat{y}_{ij} - \mu - \beta_j = \sum_{k=1}^p \xi_{ik}^* \eta_{jk}^*$$

where $\xi_{ik}^* = \lambda_k^\alpha \xi_{ik}$; $\eta_{jk}^* = \lambda_k^{1-\alpha} \eta_{jk}$ being λ_k the k th eigenvalue from the SVD ($k = 1, \dots, p$), with $p \leq \min(e, g)$; α is the singular value partition factor for the Principal Component (PC) k ; ξ_{ik}^* and η_{jk}^* are the PC scores for genotype i and environment j , respectively.

The function `gge()` from the R package `metan` Olivoto et al. (2019) according to Yan & Kang (2003) was deployed to produce the GGE model in this study.

3.9 WAASB index

The function `waasb()` function computes the Weighted Average of the Absolute Scores considering all possible IPCA from the Singular Value Decomposition of the BLUPs for genotype-vs-environment interaction effects obtained by an Linear Mixed-effect Model Olivoto et al. (2019), as follows:

$$WAASB_i = \sum_{k=1}^p |IPCA_{ik} \times EP_k| / \sum_{k=1}^p EP_k$$

where $WAASB_i$ is the weighted average of absolute scores of the i th genotype; $IPCA_{ik}$ is the scores of the i th genotype in the k th IPCA; and EP_k is the explained variance of the k th PCA for $k = 1, 2, \dots, p$, $p = \min(g - 1; e - 1)$.

3.10 WAASBY index

- Simultaneous selection for mean performance and stability

The `waasby` index is used for genotype ranking considering both the stability (`waasb`) and mean performance (`y`) based on the following model Olivoto et al. (2019).

$$waasby_i = \frac{(rY_i \times \theta_Y) + (rW_i \times \theta_W)}{\theta_Y + \theta_W}$$

where $waasby_i$ is the superiority index for the i -th genotype; rY_i and rW_i are the rescaled values (0-100) for the response variable (y) and the stability (WAAS or WAASB), respectively; θ_Y and θ_W are the weights for mean performance and stability, respectively.

4 Appendix D - R codes

4.1 MET analysis - Black beans

4.1.1 MET analysis - ASReml

Running MET using `ASReml` - only to comparison of variance components with `metan` outputs

```
mod.met.asreml.bb1 <- asreml(fixed      = yield ~ loc + loc:rep,
                             random    = ~ name + name:loc,
                             data      = blues_stage.I_BB,
                             predict   = predict.asreml(classify = "name"),
                             trace     = F,
                             maxit    = 500)

summFix.bb.met.asreml <- data.frame(wald(mod.met.asreml.bb1))
summFix.bb.met.asreml
```

Df	Sum.of.Sq	Wald.statistic	Pr.Chisq.
numeric	numeric	numeric	numeric
1	1,368,373,852	7,427.1	0
3	7,396,784	40.1	0
12	21,308,617	115.7	0
	184,240		

```
summ.bb.met.asreml<- data.frame(summary.asreml(mod.met.asreml.bb1)$varcomp)
summ.bb.met.asreml
```

component	std.error	z.ratiobound	X.ch
numeric	numeric	numericcharacter	numeric
48,605.0	17,987.9	2.7P	0
164,940.6	21,008.6	7.9P	0
184,240.9	9,279.9	19.9P	0

```
#print(summary.asreml(mod.met.asreml.bb1)$bic)
mod.met.asreml.bb<- data.table((mod.met.asreml.bb1$predictions$pvals[1:3]))
names(mod.met.asreml.bb) <- c("name", "yield_BLUPS_MET", "SE")

###
```

4.1.2 MET analysis - lme4

Running MET using **metan** R package Olivoto et al. (2019).

```
mixed_mod.bb<-
  gamem_met(blues_stage.I_BB,
    env = loc,
    gen = name,
    rep = rep,
    resp = yield,
    random = "gen", #Default
    verbose = TRUE) #Default
```

Evaluating trait yield |=====

100% 00:00:02

#> Method: REML/BLUP

#> Random effects: GEN, GEN:ENV

#> Fixed effects: ENV, REP(ENV)

```
#> Denominator DF: Satterthwaite's method
```

P-values for Likelihood Ratio Test of the analyzed traits

model yield COMPLETE NA GEN 5.40e-04 GEN:ENV 5.48e-49

All variables with significant ($p < 0.05$) genotype-vs-environment interaction

4.1.3 Printing the model outputs

4.1.3.1 Likelihood Ratio Tests The output LRT contains the Likelihood Ratio Tests for genotype and genotype-vs-environment random effects.

```
data_mod_bb_test <- get_model_data(mixed_mod.bb, "lrt")
```

```
#> Class of the model: waasb
```

```
#> Variable extracted: lrt
```

```
data_mod_bb_test
```

VAR	model	npar	logLik	AIC	LRT	Df	Pr(>Chisq)
character	character	integer	numeric	numeric	numeric	numeric	numeric
yield	GEN	18	-8,314.1	16,664.3	12.0	1	0.0
yield	GEN:ENV	18	-8,416.4	16,868.7	216.4	1	0.0

```
#customize the display of numbers and other data in a tibble
# old <- options(pillar.sigfig = 6)
#
# blues_stage.I_BB %>%
#   group_by(loc) %>%
#   dplyr::summarise(Mean = mean(yield, na.rm = TRUE))
```

```
data_mod_bb_det <- get_model_data(mixed_mod.bb, "details")
```

4.1.3.2 Detailed parameters

```
#> Class of the model: waasb
```

```
#> Variable extracted: details
```

```
data_mod_bb_det
```

Parameters	yield
character	character
Mean	3255.05
SE	19.84
SD	655
CV	20.13
Min	805.92 (B13 in HU)
Max	5310.16 (B34 in HU)
MinENV	BA (3004.17)
MaxENV	HU (3461.59)
MinGEN	B50 (2031.07)
n: 12	

4.1.3.3 Random effects The output LRT contains the Likelihood Ratio Tests for genotype and genotype-vs-environment random effects.

```
#customize the display of numbers and other data in a tibble
old <- options(pillar.sigfig = 8)

data_mod_bb_var <- get_model_data(mixed_mod.bb, "vcomp")
```

```
#> Class of the model: waasb
```

```
#> Variable extracted: vcomp
```

```
data_mod_bb_var
```

Group	yield
character	numeric
GEN	48,616.4
GEN:ENV	164,935.3
Residual	184,240.5

```
old <- options(pillar.sigfig = 4)
data_mod_bb_comp <- get_model_data(mixed_mod.bb)
```

4.1.3.4 Variance components and genetic parameters

```
#> Class of the model: waasb
```

```
#> Variable extracted: genpar
```

```
data_mod_bb_comp
```

Parameters	yield
character	numeric
Phenotypic variance	397,792.2
Heritability	0.1
GEI _r ²	0.4
h ² _{mg}	0.5
Accuracy	0.7
r _{ge}	0.5
CV _g	6.8
CV _r	13.2
CV ratio	0.5

4.1.4 MET - GGE biplot

Genotype plus Genotype-vs-Environment interaction (GGE). Mega-environment identification in multi-environment trials (MET) according to (**W?**). Yan et al. 2007

4.1.4.1 GGE ENV biplot GGE biplot done using:

- **sd**: each value is divided by the standard deviation of its corresponding environment.
- **environment**: environment-centered (G+GE)
- **environment**: singular value is entirely partitioned into the environment eigenvectors, also called column metric preserving

```
gge_model.bb <- gge(blues_stage.I_BB, loc, name, yield,
  centering = "environment", #2
  scaling = "sd", #2
  svp = "environment")#2

a <- plot(gge_model.bb, type=4,
```

```

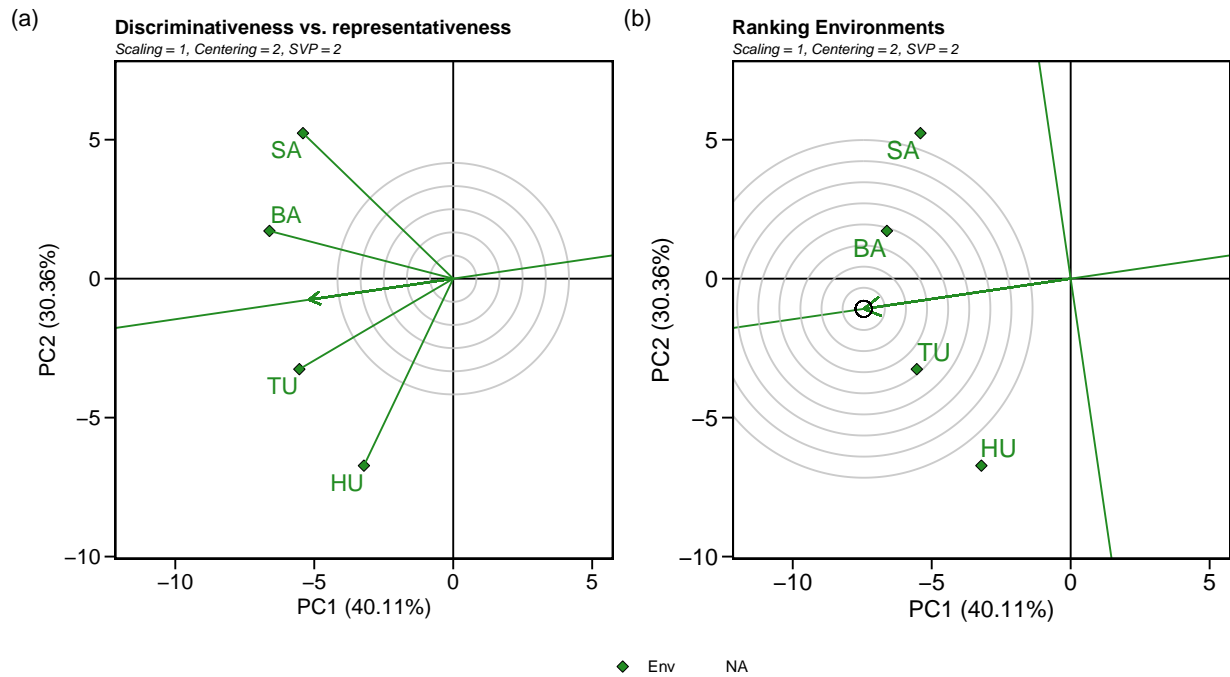
    size.text.env = 4.5,
    plot_theme = theme_metan(grid = "both",color.background =
      ↪ transparent_color()),
    axis_expand = 1.5,
    col.alpha.circle = 0.8,
    shape.gen = NA,
    col.gen = NA,
    size.text.lab = NA,
    size.text.gen = NA,
    leg.lab=c('Env'),
    #title = FALSE
  )

gge_model.bb <- gge(blues_stage.I_BB, loc, name, yield,
  centering = "environment", #1
  scaling = "sd", #2Y
  svp = "environment")#2)

b <- plot(gge_model.bb, type = 6,
  size.text.env = 5,
  plot_theme = theme_metan(grid = "both",color.background =
    ↪ transparent_color()),
  axis_expand = 1.5,
  # col.alpha.circle = 100,
  col.alpha.circle = 0.8,
  size.text.lab = 13,
  #title = FALSE
)

arrange_ggplot(a, b,
  guides = "collect",
  tag_levels = "a",
  tag_prefix = "(",
  tag_suffix = ")")

```



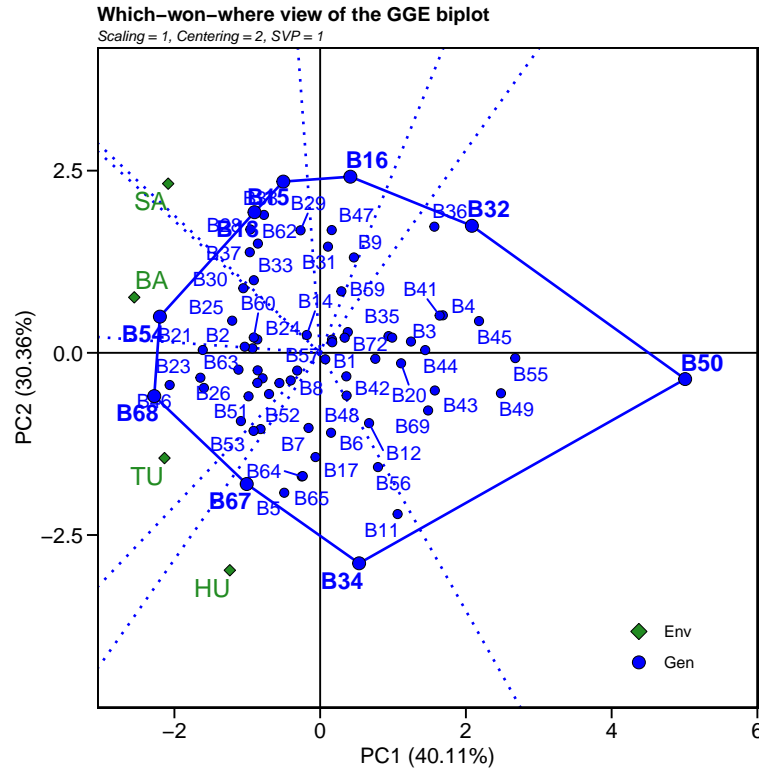
4.1.4.2 Biplot type 3: Which-won-where GGE biplot done using:

- **sd**: each value is divided by the standard deviation of its corresponding environment.
- **environment**: environment-centered (G+GE)
- **genotype**: singular value is entirely partitioned into the environment eigenvectors, also called column metric preserving

```
gge_model.bb <- gge(blues_stage.I_BB, loc, name, yield,
  centering = "environment", #2
  scaling = "sd", #1
  svp = "genotype")#2)

e <- plot(gge_model.bb, type = 3,
  size.text.env = 5,
  plot_theme = theme_metan(grid = "both", color.background =
    ↪ transparent_color()),
  axis_expand = 1.2,
  size.line = 0.7,
  size.text.gen = 4,
  size.text.win = 4.5
  #title = F
)

print(e)
```



4.1.5 Mean performance and stability analysis

WAASP index and BLUPs to estimate stability analysis.

```
#blues_stage.I_BB<- na.omit(blues_stage.I_BB)

waasb_model_bb <-
  waasb(blues_stage.I_BB,
        env = loc,
        gen = name,
        rep = rep,
        resp = yield,
        random = "gen", #Default
        verbose = TRUE,
        wresp = 60) #weight for response variable 60 and 40 for yielding and stability,
        ↪ respectively)
```

```
#> Evaluating trait yield |=====
#> -----
#> P-values for Likelihood Ratio Test of the analyzed traits
#> -----
#>      model      yield
#> COMPLETE      NA
#>      GEN 5.40e-04
#>      GEN:ENV 5.48e-49
#> -----
#> All variables with significant (p < 0.05) genotype-vs-environment interaction
```



```

waasb_model<- waasb_model_bb$yield$model

#waasb_ind <- gmd(waasb_model_bb, "WAASB")
#print_tbl(waasb_ind)

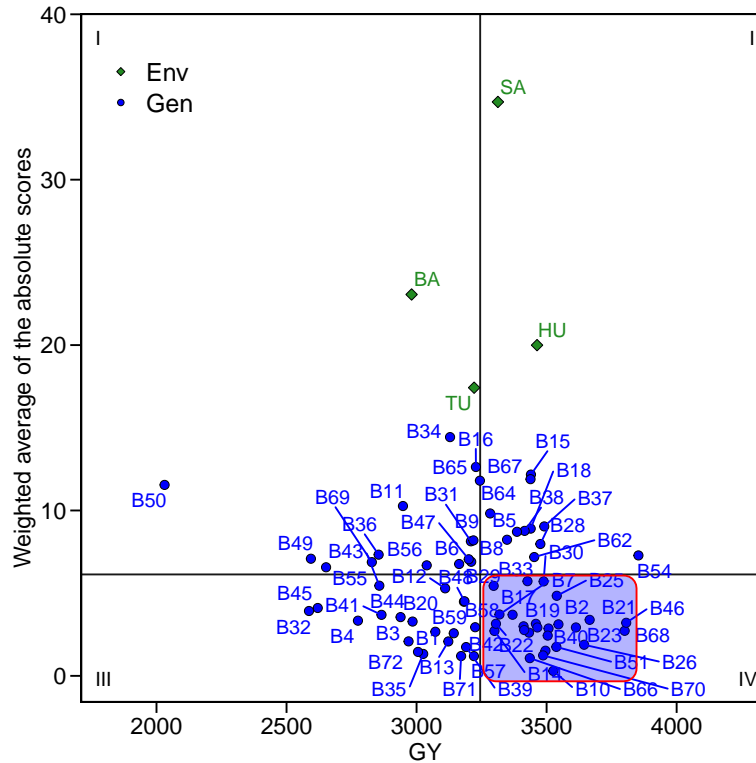
#desc <- c("Selected cultivar providing greater performance and stability for GY")

waasp_plot <- plot_scores(waasb_model_bb, type = 3,
  title = FALSE,
  size.tex.gen = 4,
  size.tex.env = 4,
  size.tex.lab = 13,
  # highlight = c("B55", "B1" , "B29", "B20" ,"B28"),
  plot_theme = theme_metan(grid = "both",color.background = transparent_color())
) +

  geom_mark_rect(aes(filter = Code %in% c("B17", "B46", "B10", "B14"),
    ),
    label.fontsize = 10,
    show.legend = F,
    con.cap = 0,
    con.colour = "red",
    color = "red",
    expand = 0.015,
    label.buffer = unit(10, "cm"))+
  #theme_gray()+
  theme(legend.position = c(0.1, 0.9),
    legend.background = element_blank(),
    legend.title = element_blank(),
    aspect.ratio = 1) +
  labs(x = "GY")

print(waasp_plot)

```



```

waasb_model_meanWaasb<-mean(waasb_model$WAASB)
waasb_model_meanY<-mean(waasb_model$Y)

selected <- waasb_model %>%
  dplyr::filter(Y >= waasb_model_meanY & WAASB <= waasb_model_meanWaasb)

selected_table <- selected

if (knitr::is_html_output()) {
  print_table(selected_table)
}else{
  selected_table[,1:8]
}

```

type	Code	Y	PC1	PC2	PC3	WAASB	PctResp
character	character	numeric	numeric	numeric	numeric	numeric	numeric
GEN	B14	3,299.5	1.7	4.6	3.1	2.7	69.6
GEN	B19	3,369.1	-0.5	-8.0	-7.6	3.7	73.4
GEN	B21	3,665.5	5.0	0.9	-1.8	3.4	89.7
n: 26							

type	Code	Y	PC1	PC2	PC3	WAASB	PctResp
character	character	numeric	numeric	numeric	numeric	numeric	numeric
GEN	B22	3,305.2	-2.3	-2.2	-7.5	3.2	69.9
GEN	B25	3,539.1	4.7	8.2	-0.2	4.8	82.8
GEN	B26	3,644.3	0.9	3.2	-3.1	1.9	88.5
GEN	B51	3,537.4	-2.4	1.3	-0.3	1.7	82.7
GEN	B52	3,458.7	-4.3	2.3	-0.5	3.1	78.4
GEN	B53	3,507.4	-3.4	2.1	2.3	2.9	81.0
n: 26							

```
#selected$Code
```

```
#Create a data frame with BLUPS - selected and non-selected
blups_sel <-
  gmd(waasb_model_bb, "blupge") %>%
  add_cols(SELECTED = ifelse(GEN %in% selected$Code, "yes", "no")) %>%
  dplyr::rename(BLUPs_sel = yield) %>%
  droplevels()

blups_sel_mean<-
  gmd(waasb_model_bb, "blupge") %>%
  add_cols(SELECTED = ifelse(GEN %in% selected$Code, "yes", "no")) %>%
  filter(SELECTED == "yes") %>%
  dplyr::summarise(mean_GY = mean(yield,na.rm = TRUE), n = n())

# Create a data frame with the waasb index - selected and non-selected
waasb_sel <-
  gmd(waasb_model_bb, "WAASB") %>%
  add_cols(SELECTED = ifelse(GEN %in% selected$Code, "yes", "no")) %>%
  dplyr::rename(WAASB_sel = yield) %>%
  droplevels()
#str(waasb_sel)

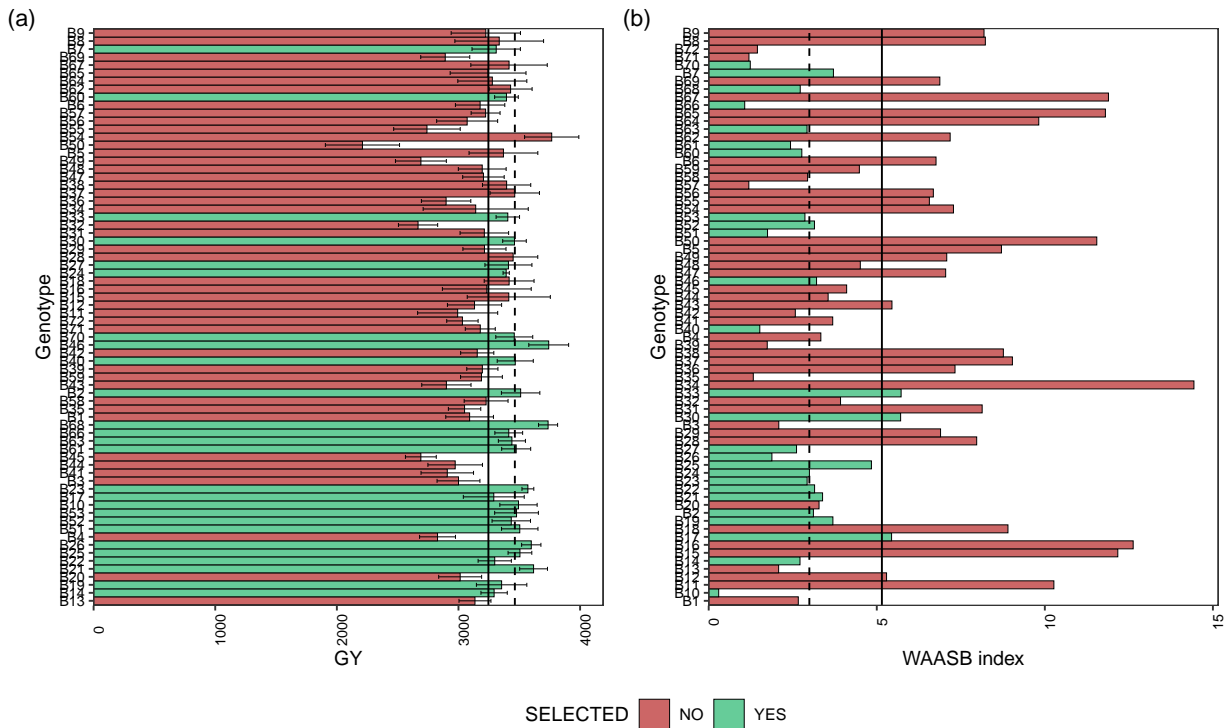
waasb_sel_mean<-
  gmd(waasb_model_bb, "WAASB") %>%
  add_cols(SELECTED = ifelse(GEN %in% selected$Code, "yes", "no")) %>%
  filter(SELECTED == "yes") %>%
  dplyr::summarise(mean_GY = mean(yield,na.rm = TRUE), n = n())

p1<- plot_selected(blups_sel, GEN, BLUPs_sel, mean_sel = blups_sel_mean$mean_GY) +
  labs(y = "GY")

p3<- plot_selected(waasb_sel, GEN, WAASB_sel, mean_sel = waasb_sel_mean$mean_GY) +
```

```
labs(y = "WAASB index")

arrange_ggplot(p1, p3,
  guides = "collect",
  tag_levels = "a",
  tag_prefix = "(",
  tag_suffix = ")")
```



Mean performance (a) and stability (b) for grain yield (GY) of 72 Black beans genotypes. The vertical dashed and solid lines shows, respectively, the mean of the selected genotype and the overall mean for both mean performance and WAASB index

4.1.5.1 Selection differentials Percentage (SD_gain in %) gain from the selected genotypes compared to the general mean.

```
blups_sel2 <-
  gmd(waasb_model_bb, "blupg") %>%
  add_cols(SELECTED = ifelse(GEN %in% selected$Code, "yes", "no")) %>%
  dplyr::rename(BLUPs_sel = yield) %>%
  droplevels()
```

```
#> Class of the model: waasb
```

```
#> Variable extracted: blupg
```

```

blups_sel_mean2<-
  gmd(waasb_model_bb, "blupg") %>%
  add_cols(SELECTED = ifelse(GEN %in% selected$Code, "yes", "no")) %>%
  filter(SELECTED == "yes") %>%
  dplyr::summarise(mean_GY = mean(yield,na.rm = TRUE), n = n())

#> Class of the model: waasb
#> Variable extracted: blupg

SD_blups<- as_tibble((blups_sel_mean2$mean_GY/mean(blups_sel2$BLUPs_sel, na.rm = T))
  ↪ -1)*100
SD_WAASP<- as_tibble((waasb_sel_mean$mean_GY /mean(waasb_sel$WAASB_sel, na.rm = T))
  ↪ -1)*100

SD_comb<- full_join(SD_blups, SD_WAASP, by = "value") %>%
  dplyr::rename(SD_gain = value) %>%
  tibble::add_column(Comp_name = c('BLUPs', 'WAASB')) %>%
  relocate(Comp_name)

SD_comb$n_selected<- blups_sel_mean2$n
SD_comb

```

Comp_name	SD_gain	n_selected
character	numeric	integer
BLUPs	3.7	26
WAASB	-41.9	26

```

blups_sel2$mean_blup <- mean(blups_sel2$BLUPs_sel, na.rm = T)
waasb_sel$mean_waasb <- mean(waasb_sel$WAASB_sel, na.rm = T)

#str(waasb_sel)
data_comb<- merge(blups_sel2, waasb_sel, by = c("GEN", "SELECTED"))
#names(data_comb)
## SD for each genotype
data_sel_perc <- data_comb %>%
  rowwise %>%
  mutate(Perc_blup_gain = ((BLUPs_sel/mean_blup)*100)-100) %>%
  mutate(Perc_WAASB_gain = ((WAASB_sel/mean_waasb)*100)-100) %>%
  as_tibble()

# data_sel_perc_mean <- data_sel_perc %>%
#   dplyr::filter(SELECTED == "yes")
#
# mean(data_sel_perc_mean$Perc_blup_gain)

if (knitr::is_html_output()) {
  print_table(data_sel_perc)
}

```

```

}else{

data_sel_perc[,1:7]
}

```

GEN	SELECTED	BLUPs_sel	mean_blup	WAASB_sel	mean_waasb	Perc_blup_gain
character	character	numeric	numeric	numeric	numeric	numeric
B1	no	3,173.6	3,255.1	2.7	5.2	-2.5
B10	yes	3,391.1	3,255.1	0.3	5.2	4.2
B11	no	3,110.5	3,255.1	10.3	5.2	-4.4
B12	no	3,185.4	3,255.1	5.3	5.2	-2.1
B13	no	3,197.5	3,255.1	2.1	5.2	-1.8
B14	yes	3,282.6	3,255.1	2.7	5.2	0.8
B15	no	3,337.8	3,255.1	12.2	5.2	2.5
B16	no	3,240.1	3,255.1	12.6	5.2	-0.5
B17	yes	3,281.1	3,255.1	5.4	5.2	0.8

n: 72

```

data_sel_perc<- data_sel_perc %>%
  dplyr::relocate(GEN,SELECTED,BLUPs_sel,mean_blup,Perc_blup_gain,
                  WAASB_sel,mean_waasb ,Perc_WAASB_gain)

#write.xlsx(data_sel_perc, "./data/sel_SD_bb_2.xlsx")

data_sel_perc2 <- data_sel_perc %>%
  dplyr::select(GEN,SELECTED, BLUPs_sel, WAASB_sel, Perc_blup_gain, Perc_WAASB_gain)

data_sel_perc2

```

GEN	SELECTED	BLUPs_sel	WAASB_sel	Perc_blup_gain	Perc_WAASB_gain
character	character	numeric	numeric	numeric	numeric
B1	no	3,173.6	2.7	-2.5	-48.3
B10	yes	3,391.1	0.3	4.2	-94.2
B11	no	3,110.5	10.3	-4.4	99.3
B12	no	3,185.4	5.3	-2.1	2.7

n: 72

GEN	SELECTED	BLUPs_sel	WAASB_sel	Perc_blup_gain	Perc_WAASB_gain
character	character	numeric	numeric	numeric	numeric
B13	no	3,197.5	2.1	-1.8	-59.6
B14	yes	3,282.6	2.7	0.8	-47.3
B15	no	3,337.8	12.2	2.5	136.2
B16	no	3,240.1	12.6	-0.5	145.1
B17	yes	3,281.1	5.4	0.8	5.6
n: 72					

```
##BLUPs indexes
stab_blups_bb<- blup_indexes(waasb_model_bb)
stab_blups_bb<- as_tibble(stab_blups_bb$yield)

data_waasby <- waasb_model_bb$yield$model %>%
  dplyr::filter(type != "ENV") %>%
  dplyr::select("Code", "WAASBY", "OrWAASBY") %>%
  dplyr::rename(GEN = Code)

stab_blups_bb<- stab_blups_bb %>%
  full_join(data_waasby, by = "GEN")

if (knitr::is_html_output()) {
  print_table(stab_blups_bb)
}else{
  stab_blups_bb[,1:8]
}
```

GEN	Y	HMGV	HMGV_R	RPGV	RPGV_Y	RPGV_R	HMRPGV
character	numeric	numeric	numeric	numeric	numeric	numeric	numeric
B1	3,122.0	3,048.9	54	1.0	3,083.2	55	0.9
B10	3,299.5	3,473.1	10	1.1	3,489.4	10	1.1
B11	3,369.1	2,884.2	62	0.9	2,986.8	61	0.9
B12	2,984.7	3,083.1	53	1.0	3,121.1	54	1.0
B13	3,665.5	3,118.1	52	1.0	3,136.3	52	1.0
B14	3,305.2	3,282.3	32	1.0	3,295.9	35	1.0
B15	3,539.1	3,310.4	30	1.1	3,419.2	23	1.0
n: 72							

GEN	Y	HMGV	HMGV_R	RPGV	RPGV_Y	RPGV_R	HMRPGV
character	numeric	numeric	numeric	numeric	numeric	numeric	numeric
B16	3,644.3	3,128.3	51	1.0	3,228.3	40	1.0
B17	2,774.6	3,232.1	35	1.0	3,278.4	37	1.0
n: 72							

```
# library(openxlsx)
# write.xlsx(stab_blups_bb, "../data/blups_bb_2.xlsx")
```

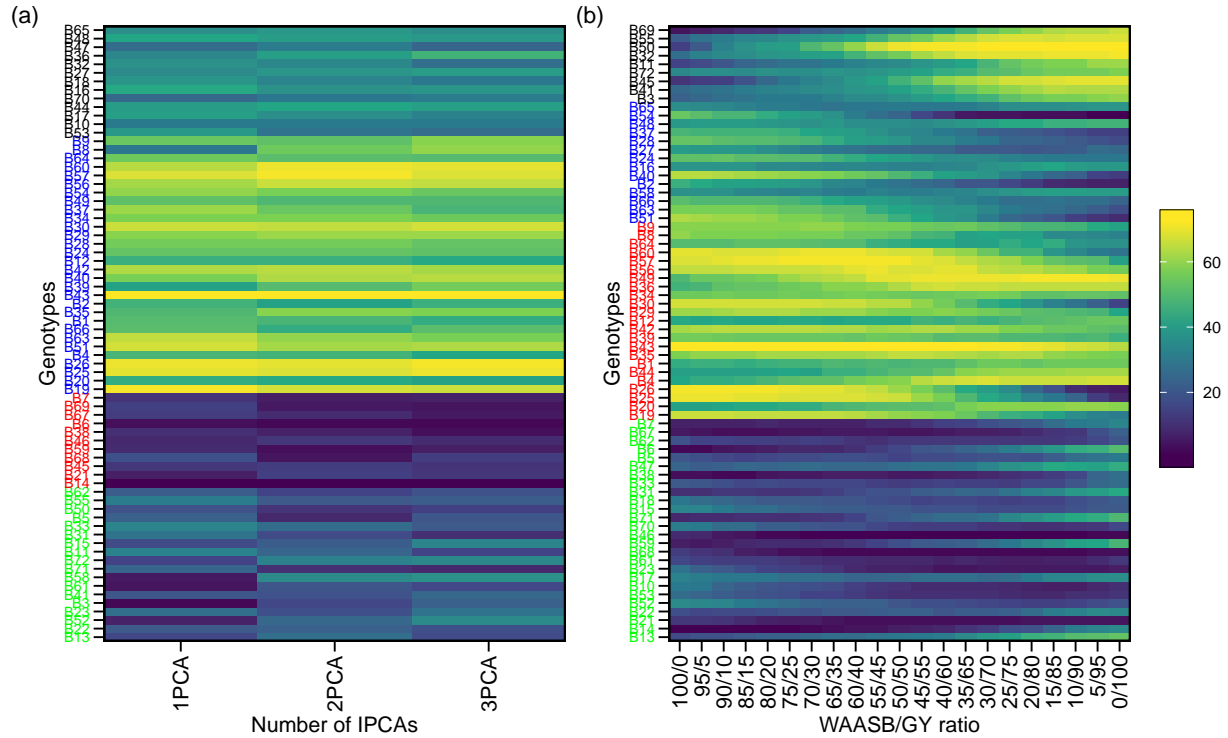
4.1.5.2 Scenarios of waasby estimation Planning different scenarios of waasby estimation by changing the weights assigned to the stability and the mean performance according to Olivoto et al. (2019)

```
scenarios <- wsmpt(waasb_model_bb, progbar = F)

scen1<- plot(scenarios, type = 1) +
  theme(axis.text.y = element_text(size=8))

scen2<-plot(scenarios, type = 2) +
  theme(axis.text.y = element_text(size=8))

arrange_ggplot(scen1, scen2,
  guides = "collect",
  legend.position = "right",
  tag_levels = "a",
  tag_prefix = "(",
  tag_suffix = ")")
```

4.1.5.3 Coincidence index of genotype selection Computes the coincidence index (Hamblin and Zimmermann, 1986) as follows:

$$CI = \frac{A - C}{M - C} \times 100$$

where A is the number of selected genotypes common to different methods; C is the number of expected genotypes selected by chance; and M is the number of genotypes selected according to the selection intensity.

```
coinc_1 <- stab_blups_bb %>% dplyr::select(GEN,HMRPGV_R) %>% arrange(HMRPGV_R)
coinc_2 <- stab_blups_bb %>% dplyr::select(GEN,RPGV_R) %>% arrange(RPGV_R)
coinc_3 <- stab_blups_bb %>% dplyr::select(GEN,HMGV_R) %>% arrange(HMGV_R)
coinc_4 <- stab_blups_bb %>% dplyr::select(GEN,OrWAASBY) %>% arrange(OrWAASBY)
coinc_5 <- stab_blups_bb %>% dplyr::select(GEN,WAASB_R) %>% arrange(WAASB_R)

selc_perc<- round(nrow(stab_blups_bb)*0.2)

coinc_1.1 <-1
coinc_1.2 <- coincidence_index(sel1 = coinc_1$GEN[1:selc_perc],
                              sel2 = coinc_2$GEN[1:selc_perc],
                              total = 72)/100
coinc_1.3 <- coincidence_index(sel1 = coinc_1$GEN[1:selc_perc],
                              sel2 = coinc_3$GEN[1:selc_perc],
                              total = 72)/100
coinc_1.4 <- coincidence_index(sel1 = coinc_1$GEN[1:selc_perc],
                              sel2 = coinc_4$GEN[1:selc_perc],
                              total = 72)/100
coinc_1.5 <- coincidence_index(sel1 = coinc_1$GEN[1:selc_perc],
```

```

                                sel2 = coinc_5$GEN[1:selc_perc],
                                total = 72)/100
coinc_2.2 <-1
coinc_2.3 <- coincidence_index(sel1 = coinc_2$GEN[1:selc_perc],
                                sel2 = coinc_3$GEN[1:selc_perc],
                                total = 72)/100
coinc_2.4 <- coincidence_index(sel1 = coinc_2$GEN[1:selc_perc],
                                sel2 = coinc_4$GEN[1:selc_perc],
                                total = 72)/100
coinc_2.5 <- coincidence_index(sel1 = coinc_2$GEN[1:selc_perc],
                                sel2 = coinc_5$GEN[1:selc_perc],
                                total = 72)/100

coinc_3.3<- 1
coinc_3.4 <- coincidence_index(sel1 = coinc_3$GEN[1:selc_perc],
                                sel2 = coinc_4$GEN[1:selc_perc],
                                total = 72)/100
coinc_3.5 <- coincidence_index(sel1 = coinc_3$GEN[1:selc_perc],
                                sel2 = coinc_5$GEN[1:selc_perc],
                                total = 72)/100

coinc_4.4 <- 1
coinc_4.5 <- coincidence_index(sel1 = coinc_4$GEN[1:selc_perc],
                                sel2 = coinc_5$GEN[1:selc_perc],
                                total = 72)/100

coinc_5.5 <- 1

coinc<- c(coinc_1.1,coinc_1.2,coinc_2.2,coinc_1.3,coinc_2.3,
          coinc_3.3,coinc_1.4, coinc_2.4, coinc_3.4,
          coinc_4.4, coinc_1.5, coinc_2.5,
          coinc_3.5, coinc_4.5,
          coinc_5.5)

z=matrix(0,5,5)
z[upper.tri(z)| row(z)==col(z)] <- coinc

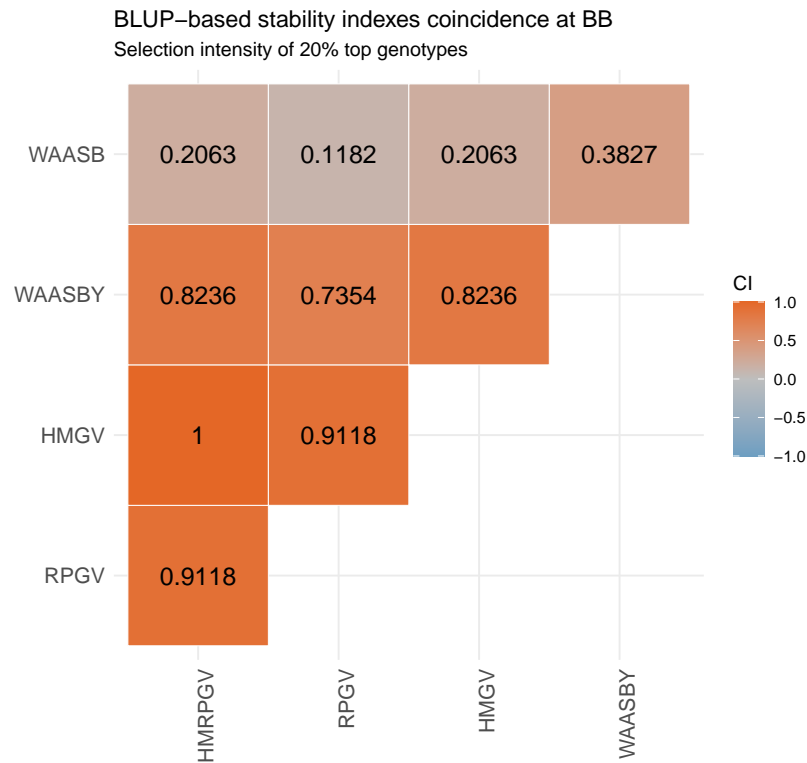
rownames(z)=c(
"HMRPGV",
"RPGV",
'HMGV',
'WAASBY',
'WAASB')

colnames(z)=rownames(z)

plotBB<- ggcorrplot(z, colors = c("#6D9EC1", "gray", "#E46726"),
                    show.legend = T,
                    legend.title = "CI", lab_size=5,tl.srt = 90,type = c("upper"), lab = T,digits = 4,
                    outline.color = "white",pch.col = "white", tl.col = "blue",show.diag = F) +
  labs(title = "BLUP-based stability indexes coincidence at BB",
        subtitle = "Selection intensity of 20% top genotypes")

print(plotBB)

```



4.2 MET analysis - Navy beans

4.2.1 MET analysis - ASReml

Running MET using ASReml - only to comparison of variance components with metan outputs

```
mod.met.asreml.nb1 <- asreml(fixed      = yield ~ loc + loc:rep,
  random      = ~ name + name:loc,
  data        = blues_stage.I_NB,
  predict     = predict.asreml(classify = "name"),
  trace       = F,
  maxit       = 500)

summFix.nb.met.asreml <- data.frame(wald(mod.met.asreml.nb1))
summFix.nb.met.asreml
```

Df	Sum.of.Sq	Wald.statistic	Pr.Chisq.
numeric	numeric	numeric	numeric
1	1,636,916,005	9,420.3	0
3	11,631,057	66.9	0
12	20,821,488	119.8	0
	173,765		

```
summ.nb.met.asreml<- data.frame(summary.asreml(mod.met.asreml.nb1)$varcomp)
summ.nb.met.asreml
```

component	std.error	z.ratiobound	X.ch
numeric	numeric	numericcharacter	numeric
34,501.8	11,258.8	3.1P	0
72,760.9	11,918.4	6.1P	0
173,765.2	8,797.2	19.8P	0

```
#print(summary.asreml(mod.met.asreml.nb1)$bic)
mod.met.asreml.nb<- data.table((mod.met.asreml.nb1$predictions$pvals[1:3]))
names(mod.met.asreml.nb) <- c("name", "yield_BLUPS_MET", "SE")

###
```

4.2.2 MET analysis - lme4

Running MET using metan R package Olivoto et al. (2019).

```
#str(blues_stage.I_NB)
mixed_mod.nb<-
  gamem_met(blues_stage.I_NB,
    env = loc,
    gen = name,
    rep = rep,
    resp = yield,
    random = "gen", #Default
    verbose = TRUE) #Default
```

Evaluating trait yield |=====

100% 00:00:02

#> Method: REML/BLUP

#> Random effects: GEN, GEN:ENV

#> Fixed effects: ENV, REP(ENV)

#> Denominator DF: Satterthwaite's method

P-values for Likelihood Ratio Test of the analyzed traits

model yield COMPLETE NA GEN 3.41e-05 GEN:ENV 6.69e-20

All variables with significant ($p < 0.05$) genotype-vs-environment interaction

4.2.3 Printing the model outputs

4.2.3.1 Likelihood Ratio Tests The output LRT contains the Likelihood Ratio Tests for genotype and genotype-vs-environment random effects.

```
data_mod_nb_test <- get_model_data(mixed_mod.nb, "lrt")
```

```
#> Class of the model: waasb
```

```
#> Variable extracted: lrt
```

```
data_mod_nb_test
```

VAR	model	npar	logLik	AIC	LRT	Df	Pr(>Chisq)
character	character	integer	numeric	numeric	numeric	numeric	numeric
yield	GEN	18	-8,123.2	16,282.4	17.2	1	0.0
yield	GEN:ENV	18	-8,156.3	16,348.7	83.4	1	0.0

```
#customize the display of numbers and other data in a tibble
# old <- options(pillar.sigfig = 6)
#
# blues_stage.I_NB %>%
#   group_by(loc) %>%
#   dplyr::summarise(Mean = mean(yield, na.rm = TRUE))
```

```
data_mod_nb_det <- get_model_data(mixed_mod.nb, "details")
```

4.2.3.2 Detailed parameters

```
#> Class of the model: waasb
```

```
#> Variable extracted: details
```

```
data_mod_nb_det
```

Parameters	yield
character	character
Mean	2924
n: 12	

Parameters	yield
character	character
SE	17.22
SD	565.35
CV	19.34
Min	1096.13 (N32 in BA)
Max	4939.4 (N34 in TU)
MinENV	BA (2654.03)
MaxENV	HU (3094)
MinGEN	N28 (2260.58)
n: 12	

4.2.3.3 Random effects The output LRT contains the Likelihood Ratio Tests for genotype and genotype-vs-environment random effects.

```
old <- options(pillar.sigfig = 8)
data_mod_nb_var <- get_model_data(mixed_mod.nb, "vcomp")
```

```
#> Class of the model: waasb
```

```
#> Variable extracted: vcomp
```

```
data_mod_nb_var
```

Group	yield
character	numeric
GEN	34,502.1
GEN:ENV	72,760.9
Residual	173,765.0

```
old <- options(pillar.sigfig = 4)
data_mod_nb_comp <- get_model_data(mixed_mod.nb)
```

4.2.3.4 Variance components and genetic parameters

```
#> Class of the model: waasb
```

```
#> Variable extracted: genpar
```

```
data_mod_nb_comp
```

Parameters	yield
character	numeric
Phenotypic variance	281,028.0
Heritability	0.1
GEIr2	0.3
h2mg	0.5
Accuracy	0.7
rge	0.3
CVg	6.4
CVr	14.3
CV ratio	0.4

4.2.4 MET - GGE biplot

Genotype plus Genotype-vs-Environment interaction (GGE). Mega-environment identification in multi-environment trials (MET) according to (**W?**). Yan et al. 2007

4.2.4.1 GGE ENV biplot GGE biplot done using:

- **sd**: each value is divided by the standard deviation of its corresponding environment.
- **environment**: environment-centered (G+GE)
- **environment**: singular value is entirely partitioned into the environment eigenvectors, also called column metric preserving

```
gge_model.nb <- gge(blues_stage.I_NB, loc, name, yield,
  centering = "environment", #1
  scaling = "sd", #2
  svp = "environment")#2

a <- plot(gge_model.nb, type=4,
  size.text.env = 4.5,
  plot_theme = theme_metan(grid = "both", color.background =
    ↪ transparent_color()),
  axis_expand = 1.5,
  col.alpha.circle = 0.8,
  shape.gen = NA,
  col.gen = NA,
  size.text.lab = NA,
```

```

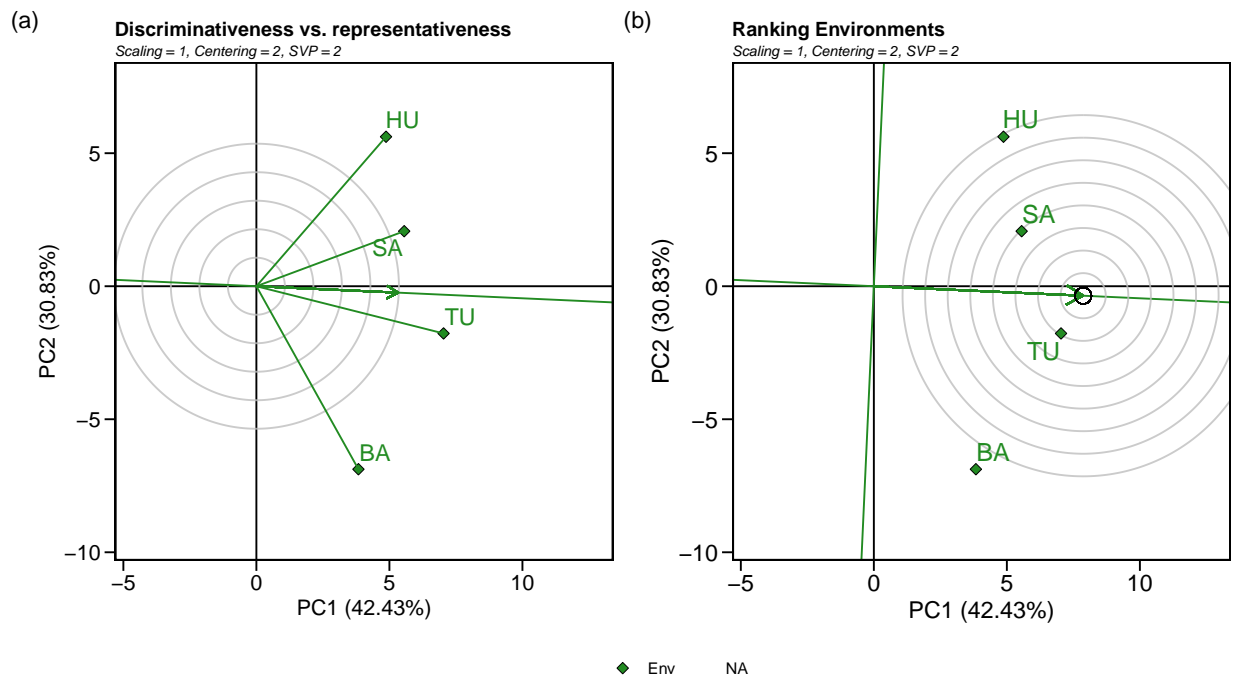
size.text.gen = NA,
leg.lab=c('Env')
#title = FALSE
)

gge_model.nb <- gge(blues_stage.I_NB, loc, name, yield,
  centering = "environment", #1
  scaling = "sd", #2Y
  svp = "environment")#2)

b <- plot(gge_model.nb, type = 6,
  size.text.env = 5,
  plot_theme = theme_metan(grid = "both", color.background =
    transparent_color()),
  axis_expand = 1.5,
  # col.alpha.circle = 100,
  col.alpha.circle = 0.8,
  size.text.lab = 13
  #title = FALSE
)

arrange_ggplot(a, b,
  guides = "collect",
  tag_levels = "a",
  tag_prefix = "(",
  tag_suffix = ")")

```



4.2.4.2 Biplot type 3: Which-won-where GGE biplot done using:

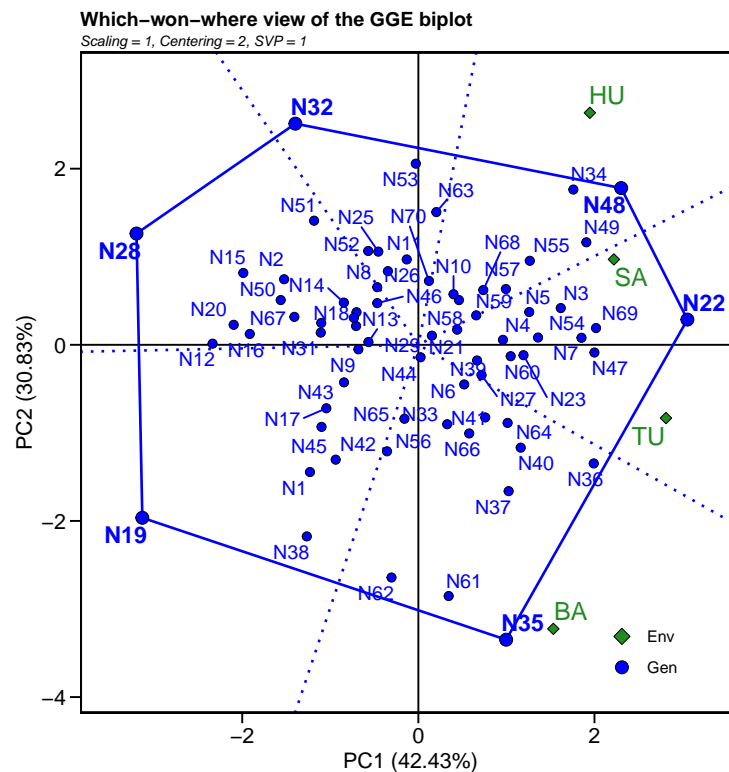
- **sd**: each value is divided by the standard deviation of its corresponding environment.
- **environment**: environment-centered (G+GE)
- **genotype**: singular value is entirely partitioned into the environment eigenvectors, also called column metric preserving

```
gge_model.nb <- gge(blues_stage.I_NB, loc, name, yield,
  centering = "environment", #2
  scaling = "sd", #1
  svp = "genotype")#2)

e <- plot(gge_model.nb, type = 3,
  size.text.env = 5,
  plot_theme = theme_metan(grid = "both", color.background =
    ↪ transparent_color()),
  axis_expand = 1.2,
  size.line = 0.7,
  size.text.gen = 4,
  size.text.win = 4.5
  #title = FALSE

)

print(e)
```



4.2.5 Mean performance and stability analysis

WAASP index and BLUPs to estimate stability analysis.

```

waasb_model_nb <-
  waasb(blues_stage.I_NB,
        env = loc,
        gen = name,
        rep = rep,
        resp = yield,
        random = "gen", #Default
        verbose = TRUE,
        wresp = 60) #weight for response variable 60 and 40 for yielding and stability,
        ↪ respectively)

```

```

#> Evaluating trait yield |=====
#> -----
#> P-values for Likelihood Ratio Test of the analyzed traits
#> -----
#>      model      yield
#> COMPLETE      NA
#>      GEN 3.41e-05
#>      GEN:ENV 6.69e-20
#> -----
#> All variables with significant (p < 0.05) genotype-vs-environment interaction

```

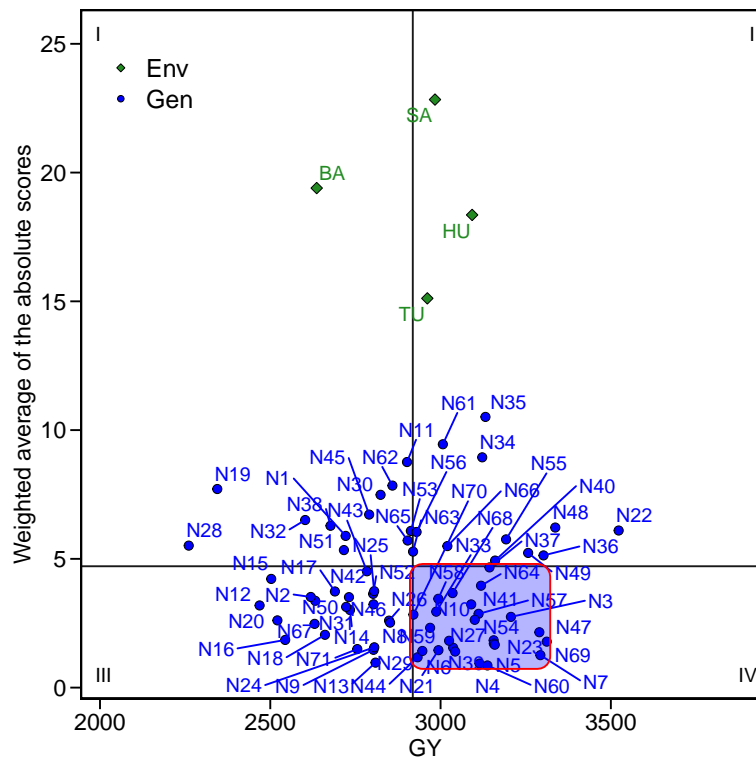
```

waasb_model<- waasb_model_nb$yield$model

waasp_plot <- plot_scores(waasb_model_nb, type = 3,
  title = FALSE,
  size.tex.gen = 4,
  size.tex.env = 4,
  size.tex.lab = 13,
  # highlight = c("N38", "N6" , "N61", "N35" ,"N52", "N22"),
  plot_theme = theme_metan(grid = "both",color.background = transparent_color())
) +

  geom_mark_rect(aes(filter = Code %in% c("N70", "N37", "N69", "N60"),
    ),
    label.fontsize = 10,
    show.legend = F,
    con.cap = 0,
    con.colour = "red",
    color = "red",
    expand = 0.005,
    label.buffer = unit(10, "cm"))+
  #theme_gray()+
  theme(legend.position = c(0.1, 0.9),
    legend.background = element_blank(),
    legend.title = element_blank(),
    aspect.ratio = 1) +
  labs(x = "GY")
print(waasp_plot)

```



```

waasb_model_meanWaasb<-mean(waasb_model$WAASB)
waasb_model_meanY<-mean(waasb_model$Y)

selected <- waasb_model %>%
  dplyr::filter(Y >= waasb_model_meanY & WAASB <= waasb_model_meanWaasb)

selected_table <- selected

if (knitr::is_html_output()) {
  print_table(selected_table)
}else{
selected_table[,1:8]
}

```

type	Code	Y	PC1	PC2	PC3	WAASB	PctResp
character	character	numeric	numeric	numeric	numeric	numeric	numeric
GEN	N37	3,143.7	-8.6	0.3	1.9	4.7	70.0
GEN	N39	3,037.0	-1.1	-1.3	3.0	1.5	61.5
GEN	N41	3,090.0	-3.0	4.8	1.4	3.2	65.7
n: 25							

type	Code	Y	PC1	PC2	PC3	WAASB	PctResp
character	character	numeric	numeric	numeric	numeric	numeric	numeric
GEN	N58	2,993.3	3.7	3.3	3.1	3.4	58.0
GEN	N64	3,118.8	-5.3	-2.7	2.7	4.0	68.0
GEN	N10	2,969.3	3.1	-0.3	3.4	2.3	56.1
GEN	N21	2,993.8	-0.1	-3.8	1.3	1.5	58.1
GEN	N33	2,987.3	-4.9	-0.9	1.3	2.9	57.6
GEN	N3	3,206.6	2.1	-1.5	6.2	2.7	74.9
n: 25							

```
#selected$Code
```

```
#Create a data frame with BLUPS - selected and non-selected
blups_sel <-
  gmd(waasb_model_nb, "blupge") %>%
  add_cols(SELECTED = ifelse(GEN %in% selected$Code, "yes", "no")) %>%
  dplyr::rename(BLUPs_sel = yield) %>%
  droplevels()

blups_sel_mean<-
  gmd(waasb_model_nb, "blupge") %>%
  add_cols(SELECTED = ifelse(GEN %in% selected$Code, "yes", "no")) %>%
  filter(SELECTED == "yes") %>%
  dplyr::summarise(mean_GY = mean(yield,na.rm = TRUE), n = n())

# Create a data frame with the waasb index - selected and non-selected
waasb_sel <-
  gmd(waasb_model_nb, "WAASB") %>%
  add_cols(SELECTED = ifelse(GEN %in% selected$Code, "yes", "no")) %>%
  dplyr::rename(WAASB_sel = yield) %>%
  droplevels()
#str(waasb_sel)

waasb_sel_mean<-
  gmd(waasb_model_nb, "WAASB") %>%
  add_cols(SELECTED = ifelse(GEN %in% selected$Code, "yes", "no")) %>%
  filter(SELECTED == "yes") %>%
  dplyr::summarise(mean_GY = mean(yield,na.rm = TRUE), n = n())

p1<- plot_selected(blups_sel, GEN, BLUPs_sel, mean_sel = blups_sel_mean$mean_GY) +
  labs(y = "GY")

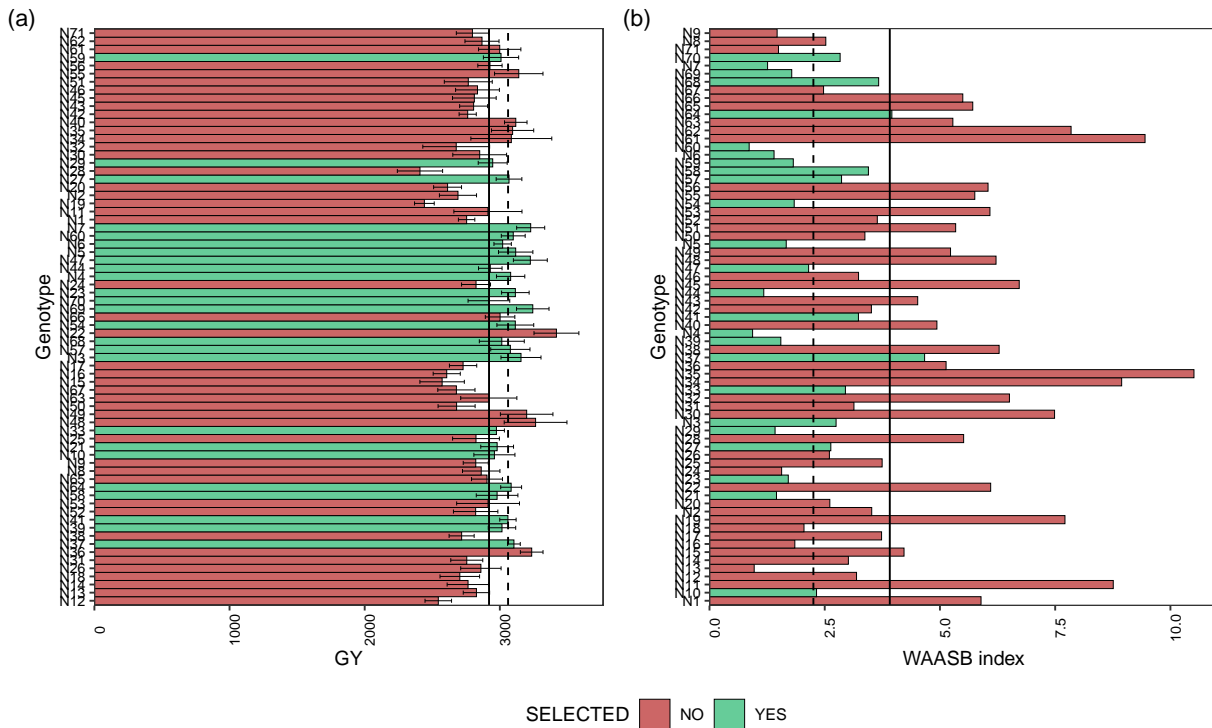
p3<- plot_selected(waasb_sel, GEN, WAASB_sel, mean_sel = waasb_sel_mean$mean_GY) +
```

```

labs(y = "WAASB index")

arrange_ggplot(p1, p3,
  guides = "collect",
  tag_levels = "a",
  tag_prefix = "(",
  tag_suffix = ")")

```



Mean performance (a) and stability (b) for grain yield (GY) of 71 Navy beans genotypes. The vertical dashed and solid lines shows, respectively, the mean of the selected genotype and the overall mean for both mean performance and WAASB index

4.2.5.1 Selection differentials Percentage (SD_gain in %) gain from the selected genotypes compared to the general mean.

```

blups_sel2 <-
  gmd(waasb_model_nb, "blupg") %>%
  add_cols(SELECTED = ifelse(GEN %in% selected$Code, "yes", "no")) %>%
  dplyr::rename(BLUPs_sel = yield) %>%
  droplevels()

```

```
#> Class of the model: waasb
```

```
#> Variable extracted: blupg
```

```

blups_sel_mean2<-
  gmd(waasb_model_nb, "blupg") %>%
  add_cols(SELECTED = ifelse(GEN %in% selected$Code, "yes", "no")) %>%
  filter(SELECTED == "yes") %>%
  dplyr::summarise(mean_GY = mean(yield,na.rm = TRUE), n = n())

#> Class of the model: waasb
#> Variable extracted: blupg

SD_blups<- as_tibble((blups_sel_mean2$mean_GY/mean(blups_sel2$BLUPs_sel, na.rm = T))
  ↪ -1)*100
SD_WAASP<- as_tibble((waasb_sel_mean$mean_GY /mean(waasb_sel$WAASB_sel, na.rm = T))
  ↪ -1)*100

SD_comb<- full_join(SD_blups, SD_WAASP, by = "value") %>%
  dplyr::rename(SD_gain = value) %>%
  tibble::add_column(Comp_name = c('BLUPs', 'WAASB')) %>%
  relocate(Comp_name)

SD_comb$n_selected<- blups_sel_mean2$n
SD_comb

```

Comp_name	SD_gain	n_selected
character	numeric	integer
BLUPs	3.2	25
WAASB	-42.4	25

```

blups_sel2$mean_blup <- mean(blups_sel2$BLUPs_sel, na.rm = T)
waasb_sel$mean_waasb <- mean(waasb_sel$WAASB_sel, na.rm = T)

#str(waasb_sel)
data_comb<- merge(blups_sel2, waasb_sel, by = c("GEN", "SELECTED"))
#names(data_comb)
## SD for each genotype
data_sel_perc <- data_comb %>%
  rowwise %>%
  mutate(Perc_blup_gain = ((BLUPs_sel/mean_blup)*100)-100) %>%
  mutate(Perc_WAASB_gain = ((WAASB_sel/mean_waasb)*100)-100) %>%
  as_tibble()

# data_sel_perc_mean <- data_sel_perc %>%
#   dplyr::filter(SELECTED == "yes")
#
# mean(data_sel_perc_mean$Perc_blup_gain)

if (knitr::is_html_output()) {
  print_table(data_sel_perc)
}

```

```

}else{

data_sel_perc[,1:7]
}

```

GEN	SELECTED	BLUPs_sel	mean_blup	WAASB_sel	mean_waasb	Perc_blup_gain
character	character	numeric	numeric	numeric	numeric	numeric
N1	no	2,820.0	2,924.0	5.9	3.9	-3.6
N10	yes	2,954.6	2,924.0	2.3	3.9	1.0
N11	no	2,899.6	2,924.0	8.8	3.9	-0.8
N12	no	2,682.8	2,924.0	3.2	3.9	-8.2
N13	no	2,867.6	2,924.0	1.0	3.9	-1.9
N14	no	2,826.9	2,924.0	3.0	3.9	-3.3
N15	no	2,701.3	2,924.0	4.2	3.9	-7.6
N16	no	2,723.6	2,924.0	1.9	3.9	-6.9
N17	no	2,802.9	2,924.0	3.7	3.9	-4.1

n: 71

```

data_sel_perc<- data_sel_perc %>%
  dplyr::relocate(GEN,SELECTED,BLUPs_sel,mean_blup,Perc_blup_gain,
                  WAASB_sel,mean_waasb ,Perc_WAASB_gain)

#write.xlsx(data_sel_perc, "./data/sel_SD_nb_2.xlsx")

data_sel_perc2 <- data_sel_perc %>%
  dplyr::select(GEN,SELECTED, BLUPs_sel, WAASB_sel, Perc_blup_gain, Perc_WAASB_gain)

data_sel_perc2

```

GEN	SELECTED	BLUPs_sel	WAASB_sel	Perc_blup_gain	Perc_WAASB_gain
character	character	numeric	numeric	numeric	numeric
N1	no	2,820.0	5.9	-3.6	50.6
N10	yes	2,954.6	2.3	1.0	-40.6
N11	no	2,899.6	8.8	-0.8	123.9
N12	no	2,682.8	3.2	-8.2	-18.4

n: 71

GEN	SELECTED	BLUPs_sel	WAASB_sel	Perc_blup_gain	Perc_WAASB_gain
character	character	numeric	numeric	numeric	numeric
N13	no	2,867.6	1.0	-1.9	-75.2
N14	no	2,826.9	3.0	-3.3	-23.0
N15	no	2,701.3	4.2	-7.6	7.9
N16	no	2,723.6	1.9	-6.9	-52.7
N17	no	2,802.9	3.7	-4.1	-4.6
n: 71					

```
##BLUPs indexes
stab_blups_nb<- blup_indexes(waasb_model_nb)
stab_blups_nb<- as_tibble(stab_blups_nb$yield)

data_waasby <- waasb_model_nb$yield$model %>%
  dplyr::filter(type != "ENV") %>%
  dplyr::select("Code", "WAASBY", "OrWAASBY") %>%
  dplyr::rename(GEN = Code)

stab_blups_nb<- stab_blups_nb %>%
  full_join(data_waasby, by = "GEN")

if (knitr::is_html_output()) {
  print_table(stab_blups_nb)
}else{
stab_blups_nb[,1:8]
}
```

GEN	Y	HMGV	HMGV_R	RPGV	RPGV_Y	RPGV_R	HMRPGV
character	numeric	numeric	numeric	numeric	numeric	numeric	numeric
N1	2,468.6	2,750.2	55	0.9	2,764.8	55	0.9
N10	2,809.0	2,933.7	33	1.0	2,954.2	32	1.0
N11	2,734.1	2,849.5	41	1.0	2,905.1	39	1.0
N12	2,660.6	2,533.6	69	0.9	2,545.2	69	0.9
N13	2,848.7	2,816.4	44	1.0	2,826.8	46	1.0
N14	2,722.7	2,739.4	57	0.9	2,761.4	57	0.9
N15	3,302.2	2,541.6	68	0.9	2,566.7	68	0.9
n: 71							

GEN	Y	HMGV	HMGV_R	RPGV	RPGV_Y	RPGV_R	HMRPGV
character	numeric	numeric	numeric	numeric	numeric	numeric	numeric
N16	3,143.7	2,595.2	67	0.9	2,606.1	67	0.9
N17	2,676.8	2,716.8	59	0.9	2,730.8	59	0.9
n: 71							

```
# library(openxlsx)
# write.xlsx(stab_blups_nb, "../data/blups_nb_2.xlsx")
```

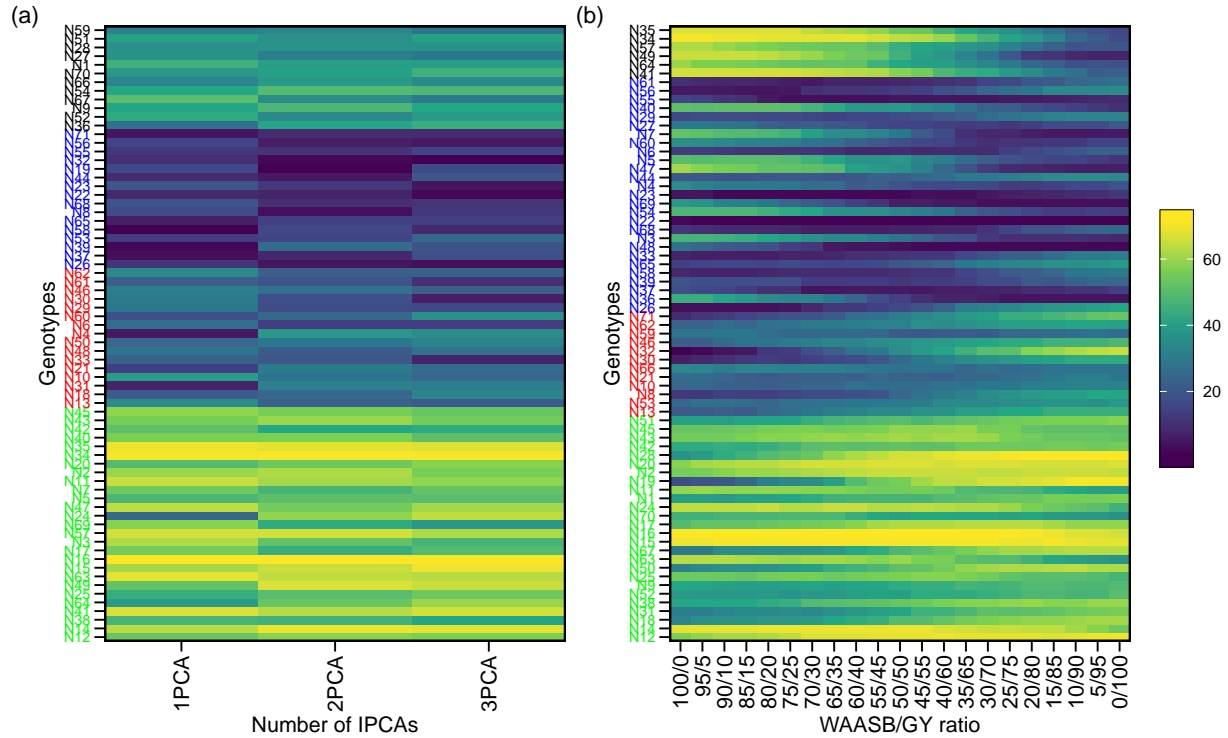
4.2.5.2 Scenarios of waasby estimation Planning different scenarios of waasby estimation by changing the weights assigned to the stability and the mean performance according to Olivoto et al. (2019).

```
scenarios <- wsmpt(waasb_model_nb, progbar = F)

scen1<- plot(scenarios, type = 1) +
  theme(axis.text.y = element_text(size=8))

scen2<-plot(scenarios, type = 2) +
  theme(axis.text.y = element_text(size=8))

arrange_ggplot(scen1, scen2,
  guides = "collect",
  legend.position = "right",
  tag_levels = "a",
  tag_prefix = "(",
  tag_suffix = ")")
```



4.2.5.3 Coincidence index of genotype selection Computes the coincidence index (Hamblin and Zimmermann, 1986) as follows:

$$CI = \frac{A - C}{M - C} \times 100$$

where A is the number of selected genotypes common to different methods; C is the number of expected genotypes selected by chance; and M is the number of genotypes selected according to the selection intensity.

```
coinc_1 <- stab_blups_nb %>% dplyr::select(GEN,HMRPGV_R) %>% arrange(HMRPGV_R)
coinc_2 <- stab_blups_nb %>% dplyr::select(GEN,RPGV_R) %>% arrange(RPGV_R)
coinc_3 <- stab_blups_nb %>% dplyr::select(GEN,HMGV_R) %>% arrange(HMGV_R)
coinc_4 <- stab_blups_nb %>% dplyr::select(GEN,OrWAASBY) %>% arrange(OrWAASBY)
coinc_5 <- stab_blups_nb %>% dplyr::select(GEN,WAASB_R) %>% arrange(WAASB_R)

selc_perc<- round(nrow(stab_blups_nb)*0.2)

coinc_1.1 <-1
coinc_1.2 <- coincidence_index(sel1 = coinc_1$GEN[1:selc_perc],
                              sel2 = coinc_2$GEN[1:selc_perc],
                              total = 71)/100
coinc_1.3 <- coincidence_index(sel1 = coinc_1$GEN[1:selc_perc],
                              sel2 = coinc_3$GEN[1:selc_perc],
                              total = 71)/100
coinc_1.4 <- coincidence_index(sel1 = coinc_1$GEN[1:selc_perc],
                              sel2 = coinc_4$GEN[1:selc_perc],
                              total = 71)/100
coinc_1.5 <- coincidence_index(sel1 = coinc_1$GEN[1:selc_perc],
```

```

                                sel2 = coinc_5$GEN[1:selc_perc],
                                total = 71)/100
coinc_2.2 <-1
coinc_2.3 <- coincidence_index(sel1 = coinc_2$GEN[1:selc_perc],
                                sel2 = coinc_3$GEN[1:selc_perc],
                                total = 71)/100
coinc_2.4 <- coincidence_index(sel1 = coinc_2$GEN[1:selc_perc],
                                sel2 = coinc_4$GEN[1:selc_perc],
                                total = 71)/100
coinc_2.5 <- coincidence_index(sel1 = coinc_2$GEN[1:selc_perc],
                                sel2 = coinc_5$GEN[1:selc_perc],
                                total = 71)/100

coinc_3.3<- 1
coinc_3.4 <- coincidence_index(sel1 = coinc_3$GEN[1:selc_perc],
                                sel2 = coinc_4$GEN[1:selc_perc],
                                total = 71)/100
coinc_3.5 <- coincidence_index(sel1 = coinc_3$GEN[1:selc_perc],
                                sel2 = coinc_5$GEN[1:selc_perc],
                                total = 71)/100

coinc_4.4 <- 1
coinc_4.5 <- coincidence_index(sel1 = coinc_4$GEN[1:selc_perc],
                                sel2 = coinc_5$GEN[1:selc_perc],
                                total = 71)/100

coinc_5.5 <- 1

coinc<- c(coinc_1.1,coinc_1.2,coinc_2.2,coinc_1.3,coinc_2.3,
          coinc_3.3,coinc_1.4, coinc_2.4, coinc_3.4,
          coinc_4.4, coinc_1.5, coinc_2.5,
          coinc_3.5, coinc_4.5,
          coinc_5.5)

z=matrix(0,5,5)
z[upper.tri(z)| row(z)==col(z)] <- coinc

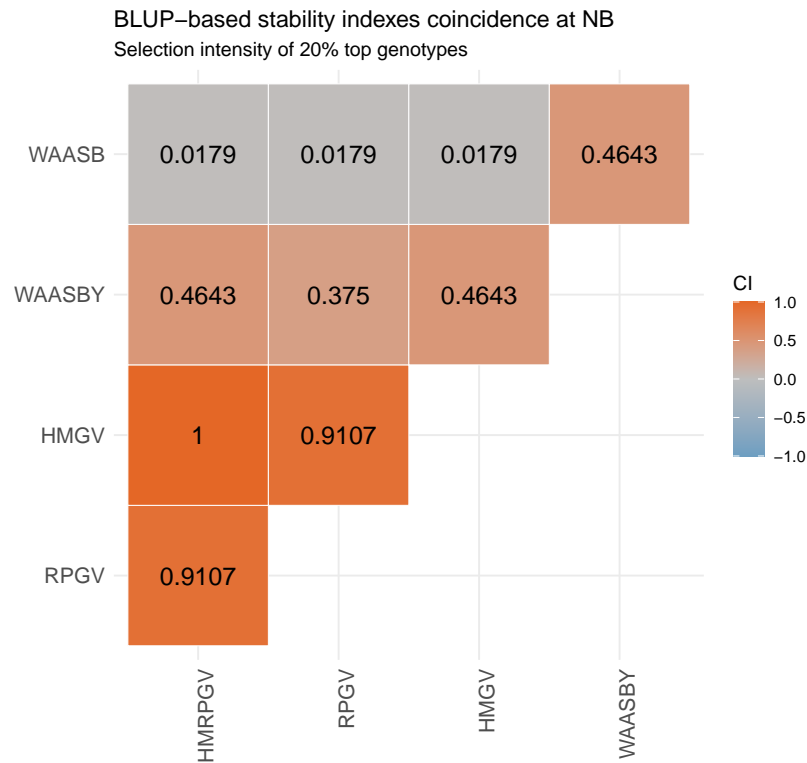
rownames(z)=c(
  "HMRPGV",
  "RPGV",
  'HMGV',
  'WAASBY',
  'WAASB')

colnames(z)=rownames(z)

plotNB<- ggcorrplot(z, colors = c("#6D9EC1", "gray", "#E46726"),
                    show.legend = T,
                    legend.title = "CI" ,lab_size=5,tl.srt = 90,type = c("upper"), lab = T,digits = 4,
                    outline.color = "white",pch.col = "white", tl.col = "blue",show.diag = F) +
  labs(title = "BLUP-based stability indexes coincidence at NB",
        subtitle = "Selection intensity of 20% top genotypes")

print(plotNB)

```



4.3 MET analysis - Red beans

4.3.1 MET analysis - ASReml

Running MET using ASReml - only to comparison of variance components with `metan` outputs

```
mod.met.asreml.sr1 <- asreml(fixed      = yield ~ loc + loc:rep,
  random      = ~ name + name:loc,
  data        = blues_stage.I_SR,
  predict     = predict.asreml(classify = "name"),
  trace       = F,
  maxit       = 500)

summFix.sr.met.asreml <- data.frame(wald(mod.met.asreml.sr1))
summFix.sr.met.asreml
```

Df	Sum.of.Sq	Wald.statistic	Pr.Chisq.
numeric	numeric	numeric	numeric
1	457,238,258	3,083.4	0.0
3	5,392,039	36.4	0.0
12	4,238,922	28.6	0.0
	148,291		

```
summ.sr.met.asreml<- data.frame(summary.asreml(mod.met.asreml.sr1)$varcomp)
summ.sr.met.asreml
```

component	std.error	z.ratiobound	X.ch
numeric	numeric	numericcharacter	numeric
31,775.8	22,260.3	1.4P	0
104,069.5	26,514.2	3.9P	0
148,291.7	13,802.9	10.7P	0

```
#print(summary.asreml(mod.met.asreml.sr1)$bic)
mod.met.asreml.sr<- data.table((mod.met.asreml.sr1$predictions$pvals[1:3]))
names(mod.met.asreml.sr) <- c("name", "yield_BLUPS_MET", "SE")

###
```

4.3.2 MET analysis - lme4

Running MET using metan R package Olivoto et al. (2019).

```
mixed_mod.sr<-
  gamem_met(blues_stage.I_SR,
    env = loc,
    gen = name,
    rep = rep,
    resp = yield,
    random = "gen", #Default
    verbose = TRUE) #Default
```

Evaluating trait yield |=====

100% 00:00:00

#> Method: REML/BLUP

#> Random effects: GEN, GEN:ENV

#> Fixed effects: ENV, REP(ENV)

#> Denominador DF: Satterthwaite's method

P-values for Likelihood Ratio Test of the analyzed traits

model yield COMPLETE NA GEN 6.80e-02 GEN:ENV 9.94e-12

All variables with significant ($p < 0.05$) genotype-vs-environment interaction

4.3.3 Printing the model outputs

4.3.3.1 Likelihood Ratio Tests The output LRT contains the Likelihood Ratio Tests for genotype and genotype-vs-environment random effects.

```
data_mod_sr_test <- get_model_data(mixed_mod.sr, "lrt")
```

```
#> Class of the model: waasb
```

```
#> Variable extracted: lrt
```

```
data_mod_sr_test
```

VAR	model	npar	logLik	AIC	LRT	Df	Pr(>Chisq)
character	character	integer	numeric	numeric	numeric	numeric	numeric
yield	GEN	18	-2,385.1	4,806.1	3.3	1	0.1
yield	GEN:ENV	18	-2,406.6	4,849.1	46.3	1	0.0

```
#customize the display of numbers and other data in a tibble
# old <- options(pillar.sigfig = 6)
#
# blues_stage.I_SR %>%
#   group_by(loc) %>%
#   dplyr::summarise(Mean = mean(yield, na.rm = TRUE))
```

```
data_mod_sr_det <- get_model_data(mixed_mod.sr, "details")
```

4.3.3.2 Detailed parameters

```
#> Class of the model: waasb
```

```
#> Variable extracted: details
```

```
data_mod_sr_det
```

Parameters	yield
character	character
Mean	3147.37
n: 12	

Parameters	yield
character	character
SE	32.29
SD	583.92
CV	18.58
Min	1325.5 (R7 in BA)
Max	4868.28 (R6 in SA)
MinENV	BA (2731.73)
MaxENV	TU (3367.38)
MinGEN	R7 (2761.81)
n: 12	

4.3.3.3 Random effects The output LRT contains the Likelihood Ratio Tests for genotype and genotype-vs-environment random effects.

```
old <- options(pillar.sigfig = 8)
data_mod_sr_var <- get_model_data(mixed_mod.sr, "vcomp")
```

```
#> Class of the model: waasb
```

```
#> Variable extracted: vcomp
```

```
data_mod_sr_var
```

Group	yield
character	numeric
GEN	31,774.8
GEN:ENV	104,069.3
Residual	148,290.9

```
old <- options(pillar.sigfig = 4)
data_mod_sr_comp <- get_model_data(mixed_mod.sr)
```

4.3.3.4 Variance components and genetic parameters

```
#> Class of the model: waasb
```

```
#> Variable extracted: genpar
```

```
data_mod_sr_comp
```

Parameters	yield
character	numeric
Phenotypic variance	284,135.1
Heritability	0.1
GEIr2	0.4
h2mg	0.5
Accuracy	0.7
rge	0.4
CVg	5.7
CVr	12.2
CV ratio	0.5

4.3.4 MET - GGE biplot

Genotype plus Genotype-vs-Environment interaction (GGE). Mega-environment identification in multi-environment trials (MET) according to (**W?**). Yan et al. 2007

4.3.4.1 GGE ENV biplot GGE biplot done using:

- **sd**: each value is divided by the standard deviation of its corresponding environment.
- **environment**: environment-centered (G+GE)
- **environment**: singular value is entirely partitioned into the environment eigenvectors, also called column metric preserving

```
gge_model.sr <- gge(blues_stage.I_SR, loc, name, yield,
  centering = "environment", #1
  scaling = "sd", #2
  svp = "environment")#2

a <- plot(gge_model.sr, type=4,
  size.text.env = 4.5,
  plot_theme = theme_metan(grid = "both",color.background =
    ↪ transparent_color()),
  axis_expand = 1.5,
  col.alpha.circle = 0.8,
  shape.gen = NA,
  col.gen = NA,
  size.text.lab = NA,
```



```

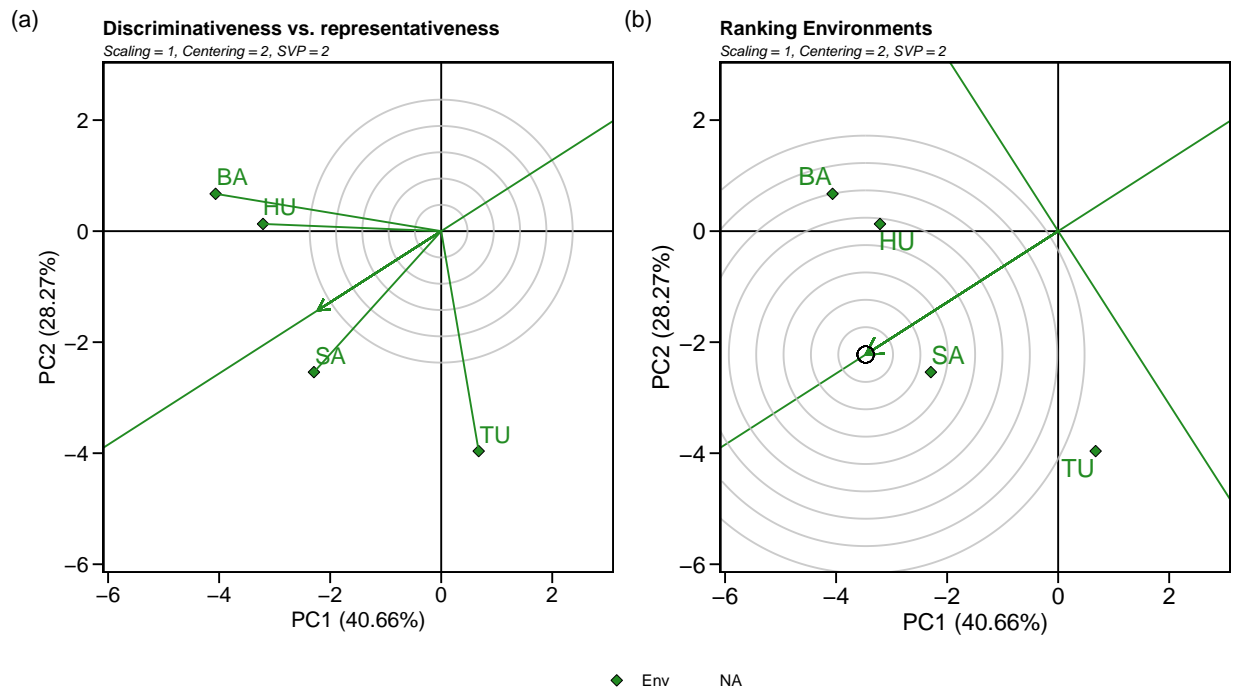
size.text.gen = NA,
leg.lab=c('Env'),
#title = FALSE
)

gge_model.sr <- gge(blues_stage.I_SR, loc, name, yield,
  centering = "environment", #1
  scaling = "sd", #2Y
  svp = "environment")#2)

b <- plot(gge_model.sr, type = 6,
  size.text.env = 5,
  plot_theme = theme_metan(grid = "both", color.background =
    ↪ transparent_color()),
  axis_expand = 1.5,
  # col.alpha.circle = 100,
  col.alpha.circle = 0.8,
  size.text.lab = 13,
  #title = FALSE
)

arrange_ggplot(a, b,
  guides = "collect",
  tag_levels = "a",
  tag_prefix = "(",
  tag_suffix = ")")

```



4.3.4.2 Biplot type 3: Which-won-where GGE biplot done using:

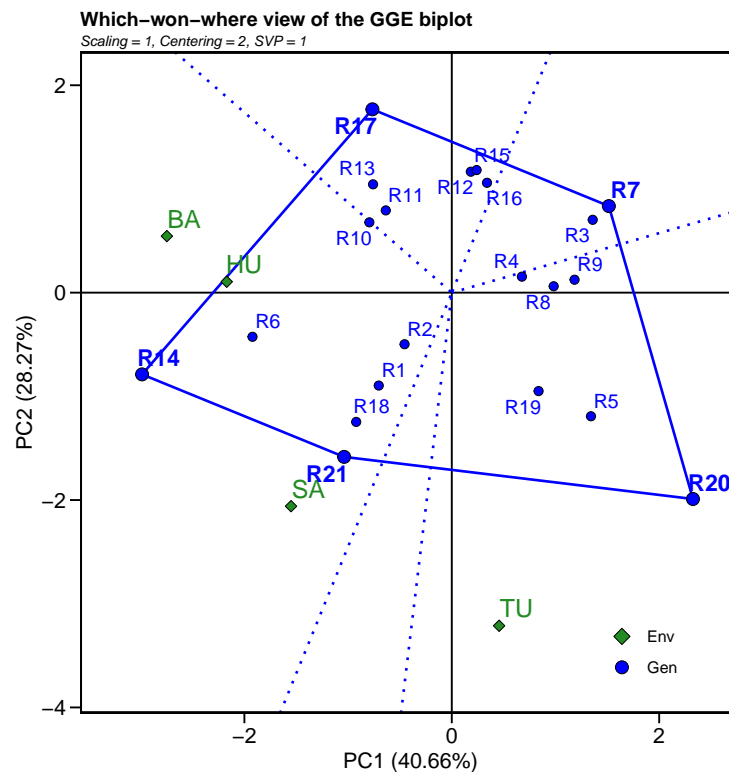
- **sd**: each value is divided by the standard deviation of its corresponding environment.
- **environment**: environment-centered (G+GE)
- **genotype**: singular value is entirely partitioned into the environment eigenvectors, also called column metric preserving

```
gge_model.sr <- gge(blues_stage.I_SR, loc, name, yield,
  centering = "environment", #2
  scaling = "sd", #1
  svp = "genotype")#2)

e <- plot(gge_model.sr, type = 3,
  size.text.env = 5,
  plot_theme = theme_metan(grid = "both", color.background =
    ↪ transparent_color()),
  axis_expand = 1.2,
  size.line = 0.7,
  size.text.gen = 4,
  size.text.win = 4.5
  #title = FALSE

)

print(e)
```



4.3.5 Mean performance and stability analysis

WAASP index and BLUPs to estimate stability analysis.

```

waasb_model_sr <-
  waasb(blues_stage.I_SR,
        env = loc,
        gen = name,
        rep = rep,
        resp = yield,
        random = "gen", #Default
        verbose = TRUE,
        wresp = 60) #weight for response variable 60 and 40 for yielding and stability,
        ↪ respectively)

```

```

#> Evaluating trait yield |=====
#> -----
#> P-values for Likelihood Ratio Test of the analyzed traits
#> -----
#>      model      yield
#> COMPLETE      NA
#>      GEN 6.80e-02
#>      GEN:ENV 9.94e-12
#> -----
#> All variables with significant (p < 0.05) genotype-vs-environment interaction

```

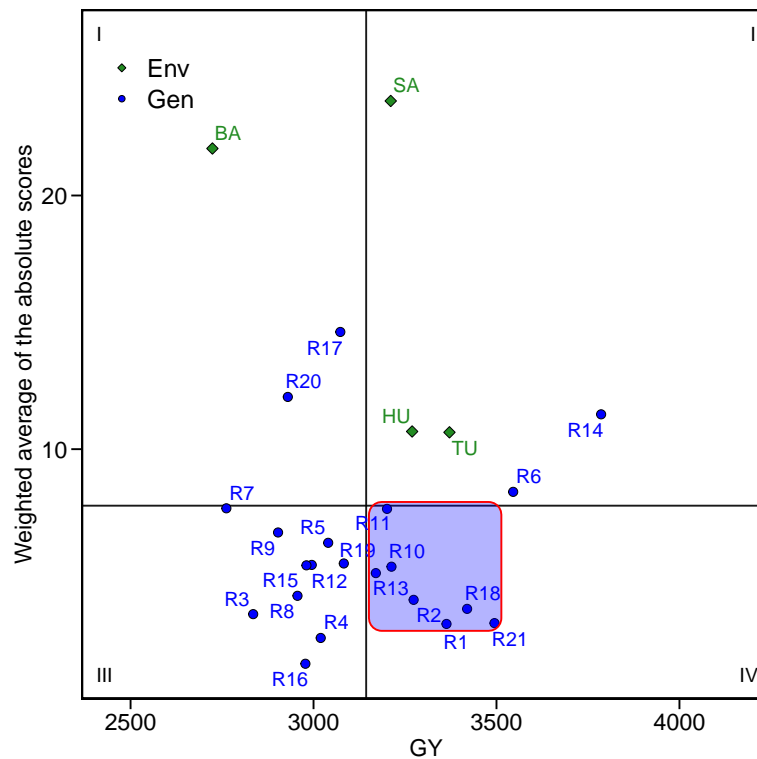
```

waasb_model<- waasb_model_sr$yield$model

waasp_plot <- plot_scores(waasb_model_sr, type = 3,
  title = FALSE,
  size.tex.gen = 4,
  size.tex.env = 4,
  size.tex.lab = 13,
  #highlight = c('R13','R2'),
  plot_theme = theme_metan(grid = "both",color.background = transparent_color())
) +

  geom_mark_rect(aes(filter = Code %in% c("R13", "R11", "R21", "R1"),
    ),
    label.fontsize = 10,
    show.legend = F,
    con.cap = 0,
    con.colour = "red",
    color = "red",
    expand = 0.01,
    label.buffer = unit(10, "cm"))+
#theme_gray()+
theme(legend.position = c(0.1, 0.9),
  legend.background = element_blank(),
  legend.title = element_blank(),
  aspect.ratio = 1) +
  labs(x = "GY")
print(waasp_plot)

```



```

waasb_model_meanWaasb<-mean(waasb_model$WAASB)
waasb_model_meanY<-mean(waasb_model$Y)

selected <- waasb_model %>%
  dplyr::filter(Y >= waasb_model_meanY & WAASB <= waasb_model_meanWaasb)

selected_table <- selected

if (knitr::is_html_output()) {
  print_table(selected_table)
}else{
selected_table[,1:8]
}

```

type	Code	Y	PC1	PC2	PC3	WAASB	PctResp
character	character	numeric	numeric	numeric	numeric	numeric	numeric
GEN	R10	3,213.0	-7.5	-2.3	7.0	5.4	44.1
GEN	R13	3,170.3	-0.8	11.6	-0.3	5.1	39.9
GEN	R18	3,419.7	1.3	-3.4	14.3	3.7	64.2
GEN	R1	3,363.4	-3.2	-2.7	4.2	3.1	58.7

type	Code	Y	PC1	PC2	PC3	WAASB	PctResp
character	character	numeric	numeric	numeric	numeric	numeric	numeric
GEN	R2	3,273.6	3.2	6.0	0.9	4.1	50.0
GEN	R21	3,494.5	-3.0	-2.6	5.4	3.1	71.5
GEN	R11	3,200.8	-13.6	-0.6	-7.8	7.6	42.9

```
#selected$Code
```

```
#Create a data frame with BLUPS - selected and non-selected
blups_sel <-
  gmd(waasb_model_sr, "blupge") %>%
  add_cols(SELECTED = ifelse(GEN %in% selected$Code, "yes", "no")) %>%
  dplyr::rename(BLUPs_sel = yield) %>%
  droplevels()

blups_sel_mean<-
  gmd(waasb_model_sr, "blupge") %>%
  add_cols(SELECTED = ifelse(GEN %in% selected$Code, "yes", "no")) %>%
  filter(SELECTED == "yes") %>%
  dplyr::summarise(mean_GY = mean(yield,na.rm = TRUE), n = n())

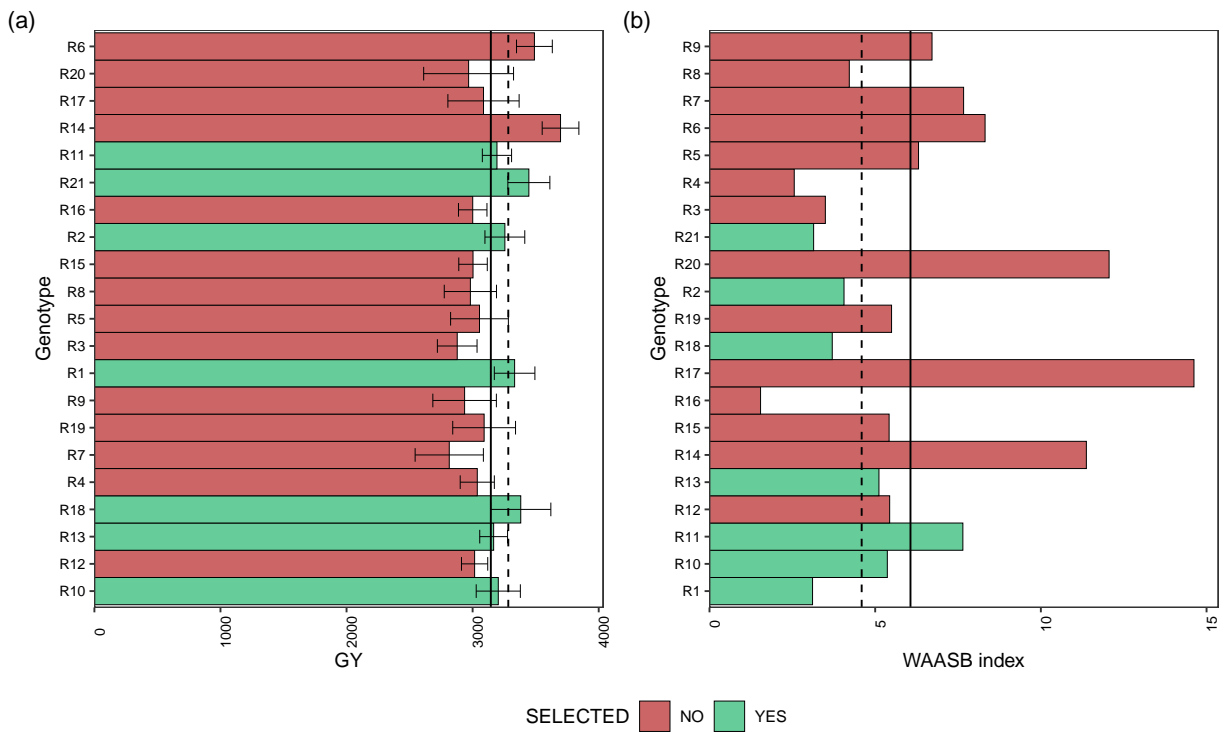
# Create a data frame with the waasb index - selected and non-selected
waasb_sel <-
  gmd(waasb_model_sr, "WAASB") %>%
  add_cols(SELECTED = ifelse(GEN %in% selected$Code, "yes", "no")) %>%
  dplyr::rename(WAASB_sel = yield) %>%
  droplevels()
#str(waasb_sel)

waasb_sel_mean<-
  gmd(waasb_model_sr, "WAASB") %>%
  add_cols(SELECTED = ifelse(GEN %in% selected$Code, "yes", "no")) %>%
  filter(SELECTED == "yes") %>%
  dplyr::summarise(mean_GY = mean(yield,na.rm = TRUE), n = n())

p1<- plot_selected(blups_sel, GEN, BLUPs_sel, mean_sel = blups_sel_mean$mean_GY) +
  labs(y = "GY")

p3<- plot_selected(waasb_sel, GEN, WAASB_sel, mean_sel = waasb_sel_mean$mean_GY) +
  labs(y = "WAASB index")

arrange_ggplot(p1, p3,
  guides = "collect",
  tag_levels = "a",
  tag_prefix = "(",
  tag_suffix = ")")
```



Mean performance (a) and stability (b) for grain yield (GY) of 24 Small Red beans genotypes. The vertical dashed and solid lines shows, respectively, the mean of the selected genotype and the overall mean for both mean performance and WAASB index

4.3.5.1 Selection differentials Percentage (SD_gain in %) gain from the selected genotypes compared to the general mean.

```
blups_sel2 <-
  gmd(waasb_model_sr, "blupg") %>%
  add_cols(SELECTED = ifelse(GEN %in% selected$Code, "yes", "no")) %>%
  dplyr::rename(BLUPs_sel = yield) %>%
  droplevels()
```

```
#> Class of the model: waasb
```

```
#> Variable extracted: blupg
```

```
blups_sel_mean2<-
  gmd(waasb_model_sr, "blupg") %>%
  add_cols(SELECTED = ifelse(GEN %in% selected$Code, "yes", "no")) %>%
  filter(SELECTED == "yes") %>%
  dplyr::summarise(mean_GY = mean(yield, na.rm = TRUE), n = n())
```

```
#> Class of the model: waasb
```

```
#> Variable extracted: blupg
```

```
SD_blups<- as_tibble((blups_sel_mean2$mean_GY/mean(blups_sel2$BLUPs_sel, na.rm = T))
  ↪ -1)*100
SD_WAASP<- as_tibble((waasb_sel_mean$mean_GY /mean(waasb_sel$WAASB_sel, na.rm = T))
  ↪ -1)*100

SD_comb<- full_join(SD_blups, SD_WAASP, by = "value") %>%
  dplyr::rename(SD_gain = value) %>%
  tibble::add_column(Comp_name = c('BLUPs', 'WAASB')) %>%
  relocate(Comp_name)

SD_comb$n_selected<- blups_sel_mean2$n
SD_comb
```

Comp_name	SD_gain	n_selected
character	numeric	integer
BLUPs	2.4	7
WAASB	-24.3	7

```
blups_sel2$mean_blup <- mean(blups_sel2$BLUPs_sel, na.rm = T)
waasb_sel$mean_waasb <- mean(waasb_sel$WAASB_sel, na.rm = T)

#str(waasb_sel)
data_comb<- merge(blups_sel2, waasb_sel, by = c("GEN", "SELECTED"))
#names(data_comb)
## SD for each genotype
```

```

data_sel_perc <- data_comb %>%
  rowwise %>%
  mutate(Perc_blup_gain = ((BLUPs_sel/mean_blup)*100)-100) %>%
  mutate(Perc_WAASB_gain = ((WAASB_sel/mean_waasb)*100)-100) %>%
  as_tibble()

# data_sel_perc_mean <- data_sel_perc %>%
#   dplyr::filter(SELECTED == "yes")
#
# mean(data_sel_perc_mean$Perc_blup_gain)

if (knitr::is_html_output()) {
  print_table(data_sel_perc)
}else{
data_sel_perc[,1:7]
}

```

GEN	SELECTED	BLUPs_sel	mean_blup	WAASB_sel	mean_waasb	Perc_blup_gain
character	character	numeric	numeric	numeric	numeric	numeric
R1	yes	3,251.2	3,147.4	3.1	6.1	3.3
R10	yes	3,180.0	3,147.4	5.4	6.1	1.0
R11	yes	3,174.2	3,147.4	7.6	6.1	0.9
R12	no	3,076.7	3,147.4	5.4	6.1	-2.2
R13	yes	3,159.7	3,147.4	5.1	6.1	0.4
R14	no	3,451.4	3,147.4	11.4	6.1	9.7
R15	no	3,069.8	3,147.4	5.4	6.1	-2.5
R16	no	3,068.5	3,147.4	1.5	6.1	-2.5
R17	no	3,113.8	3,147.4	14.6	6.1	-1.1

n: 21

```

data_sel_perc<- data_sel_perc %>%
  dplyr::relocate(GEN,SELECTED,BLUPs_sel,mean_blup,Perc_blup_gain,
    WAASB_sel,mean_waasb ,Perc_WAASB_gain)

#write.xlsx(data_sel_perc, "./data/sel_SD_sr_2.xlsx")

data_sel_perc2 <- data_sel_perc %>%
  dplyr::select(GEN,SELECTED, BLUPs_sel, WAASB_sel, Perc_blup_gain, Perc_WAASB_gain)

data_sel_perc2

```


GEN	SELECTED	BLUPs_sel	WAASB_sel	Perc_blup_gain	Perc_WAASB_gain
character	character	numeric	numeric	numeric	numeric
R1	yes	3,251.2	3.1	3.3	-48.7
R10	yes	3,180.0	5.4	1.0	-11.5
R11	yes	3,174.2	7.6	0.9	26.1
R12	no	3,076.7	5.4	-2.2	-10.4
R13	yes	3,159.7	5.1	0.4	-15.7
R14	no	3,451.4	11.4	9.7	87.5
R15	no	3,069.8	5.4	-2.5	-10.6
R16	no	3,068.5	1.5	-2.5	-74.6
R17	no	3,113.8	14.6	-1.1	141.0
n: 21					

```
##BLUPs indexes
stab_blups_sr<- blup_indexes(waasb_model_sr)
stab_blups_sr<- as_tibble(stab_blups_sr$yield)

data_waasby <- waasb_model_sr$yield$model %>%
  dplyr::filter(type != "ENV") %>%
  dplyr::select("Code", "WAASBY", "OrWAASBY") %>%
  dplyr::rename(GEN = Code)

stab_blups_sr<- stab_blups_sr %>%
  full_join(data_waasby, by = "GEN")

if (knitr::is_html_output()) {
  print_table(stab_blups_sr)
}else{
  stab_blups_sr[,1:8]
}
```

GEN	Y	HMGV	HMGV_R	RPGV	RPGV_Y	RPGV_R	HMRPGV
character	numeric	numeric	numeric	numeric	numeric	numeric	numeric
R1	3,213.0	3,307.4	5	1.1	3,332.3	5	1.1
R10	2,995.1	3,173.5	8	1.0	3,205.9	8	1.0
n: 21							

GEN	Y	HMGV	HMGV_R	RPGV	RPGV_Y	RPGV_R	HMRPGV
character	numeric	numeric	numeric	numeric	numeric	numeric	numeric
R11	3,170.3	3,181.2	7	1.0	3,211.9	7	1.0
R12	3,419.7	3,005.0	13	1.0	3,027.2	14	1.0
R13	3,019.6	3,155.4	9	1.0	3,179.9	9	1.0
R14	2,761.8	3,681.0	1	1.2	3,723.7	1	1.2
R15	3,082.8	2,990.4	15	1.0	3,012.4	15	1.0
R16	2,903.1	2,987.0	16	1.0	3,005.0	16	1.0
R17	3,363.4	3,005.7	12	1.0	3,095.9	10	1.0
n: 21							

```
# library(openxlsx)
# write.xlsx(stab_blups_sr, "../data/blups_sr_2.xlsx")
```

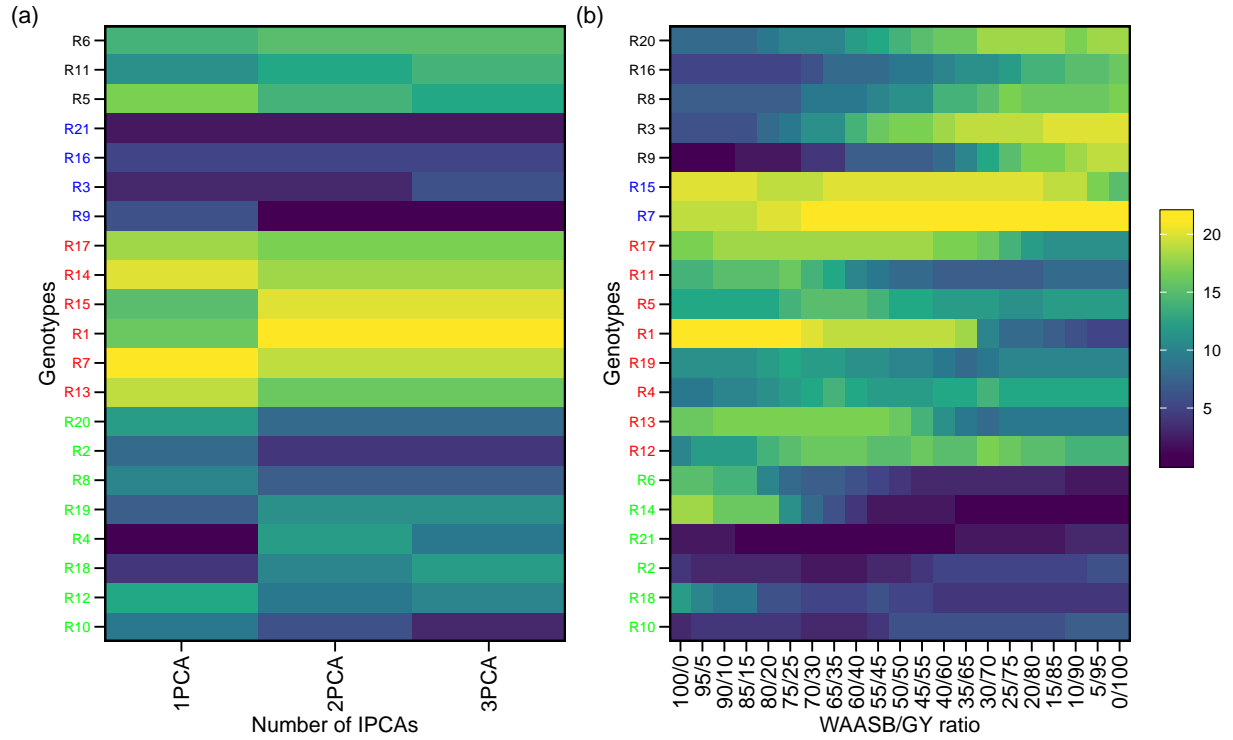
4.3.5.2 Scenarios of waasby estimation Planning different scenarios of waasby estimation by changing the weights assigned to the stability and the mean performance according to Olivoto et al. (2019).

```
scenarios <- wsmpt(waasb_model_sr, progbar = F)

scen1<- plot(scenarios, type = 1) +
  theme(axis.text.y = element_text(size=8))

scen2<-plot(scenarios, type = 2) +
  theme(axis.text.y = element_text(size=8))

arrange_ggplot(scen1, scen2,
  guides = "collect",
  legend.position = "right",
  tag_levels = "a",
  tag_prefix = "(",
  tag_suffix = ")")
```



```
#plot_eigen(waasb_model_sr, size.lab = 14, size.tex.lab = 14)
```

4.3.5.3 Coincidence index of genotype selection Computes the coincidence index (Hamblin and Zimmermann, 1986) as follows:

$$CI = \frac{A - C}{M - C} \times 100$$

where A is the number of selected genotypes common to different methods; C is the number of expected genotypes selected by chance; and M is the number of genotypes selected according to the selection intensity.

```
coinc_1 <- stab_blups_sr %>% dplyr::select(GEN,HMRPGV_R) %>% arrange(HMRPGV_R)
coinc_2 <- stab_blups_sr %>% dplyr::select(GEN,RPGV_R) %>% arrange(RPGV_R)
coinc_3 <- stab_blups_sr %>% dplyr::select(GEN,HMGV_R) %>% arrange(HMGV_R)
coinc_4 <- stab_blups_sr %>% dplyr::select(GEN,OrWAASBY) %>% arrange(OrWAASBY)
coinc_5 <- stab_blups_sr %>% dplyr::select(GEN,WAASB_R) %>% arrange(WAASB_R)

selc_perc<- round(nrow(stab_blups_sr)*0.2)

coinc_1.1 <-1
coinc_1.2 <- coincidence_index(sel1 = coinc_1$GEN[1:selc_perc],
                              sel2 = coinc_2$GEN[1:selc_perc],
                              total = 21)/100
coinc_1.3 <- coincidence_index(sel1 = coinc_1$GEN[1:selc_perc],
                              sel2 = coinc_3$GEN[1:selc_perc],
                              total = 21)/100
coinc_1.4 <- coincidence_index(sel1 = coinc_1$GEN[1:selc_perc],
```

```

                                sel2 = coinc_4$GEN[1:selc_perc],
                                total = 21)/100
coinc_1.5 <- coincidence_index(sel1 = coinc_1$GEN[1:selc_perc],
                                sel2 = coinc_5$GEN[1:selc_perc],
                                total = 21)/100

coinc_2.2 <- 1
coinc_2.3 <- coincidence_index(sel1 = coinc_2$GEN[1:selc_perc],
                                sel2 = coinc_3$GEN[1:selc_perc],
                                total = 21)/100
coinc_2.4 <- coincidence_index(sel1 = coinc_2$GEN[1:selc_perc],
                                sel2 = coinc_4$GEN[1:selc_perc],
                                total = 21)/100
coinc_2.5 <- coincidence_index(sel1 = coinc_2$GEN[1:selc_perc],
                                sel2 = coinc_5$GEN[1:selc_perc],
                                total = 21)/100

coinc_3.3 <- 1
coinc_3.4 <- coincidence_index(sel1 = coinc_3$GEN[1:selc_perc],
                                sel2 = coinc_4$GEN[1:selc_perc],
                                total = 21)/100
coinc_3.5 <- coincidence_index(sel1 = coinc_3$GEN[1:selc_perc],
                                sel2 = coinc_5$GEN[1:selc_perc],
                                total = 21)/100

coinc_4.4 <- 1
coinc_4.5 <- coincidence_index(sel1 = coinc_4$GEN[1:selc_perc],
                                sel2 = coinc_5$GEN[1:selc_perc],
                                total = 21)/100

coinc_5.5 <- 1

coinc <- c(coinc_1.1,coinc_1.2,coinc_2.2,coinc_1.3,coinc_2.3,
           coinc_3.3,coinc_1.4, coinc_2.4, coinc_3.4,
           coinc_4.4, coinc_1.5, coinc_2.5,
           coinc_3.5, coinc_4.5,
           coinc_5.5)

z=matrix(0,5,5)
z[upper.tri(z)| row(z)==col(z)] <- coinc

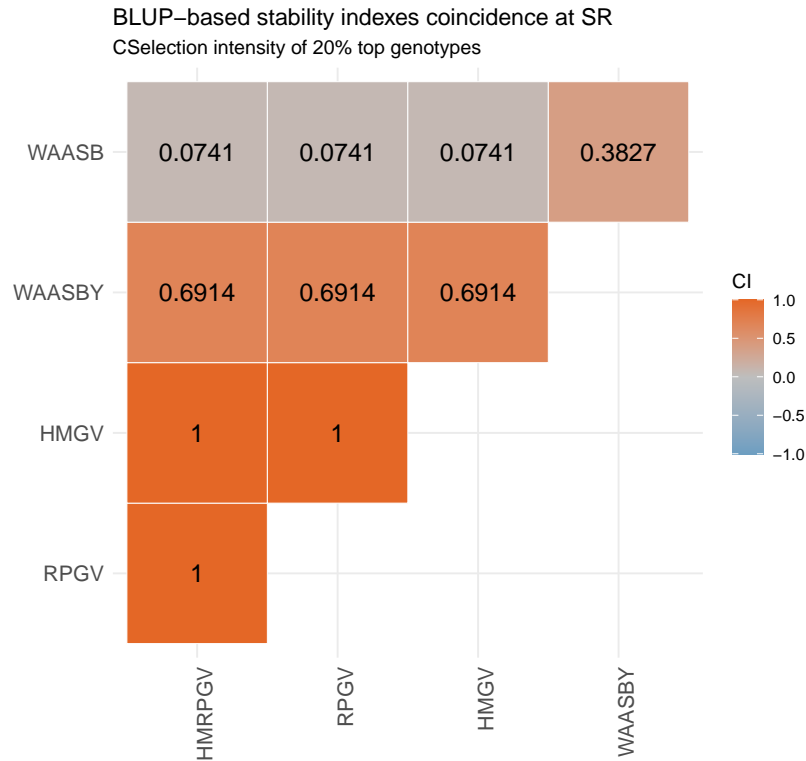
rownames(z)=c(
  "HMRPGV",
  "RPGV",
  'HMGV',
  'WAASBY',
  'WAASB')

colnames(z)=rownames(z)

plotSR<- ggcorrplot(z, colors = c("#6D9EC1", "gray", "#E46726"),
                    show.legend = T,
                    legend.title = "CI" ,lab_size=5,tl.srt = 90,type = c("upper"), lab = T,digits = 4,
                    outline.color = "white",pch.col = "white", tl.col = "blue",show.diag = F) +
  labs(title = "BLUP-based stability indexes coincidence at SR",
        subtitle = "CSelection intensity of 20% top genotypes")

```

```
print(plotSR)
```



5 Appendix E - R codes

5.1 Multi-trait Multi-env analysis

5.2 Descriptive Stats - Multi-trait data

- GY data set: were obtained from the adjusted data in the 2nd-stage mixed models analysis.
- PH, PM, and LD were obtained from the coincident locations (TU, BA and SA) in 2021 only.

5.2.1 Data set

```
data_beans_compl = read.csv("data/data_beans_Multi-trait.csv",h=T, stringsAsFactors = T)
data_beans_compl<- data_beans_compl[,-c(1:2)]
colnames(data_beans_compl)<- c('name', 'loc', 'mkt', 'rep', 'GY', 'DM', 'PH', "LD")
#str(data_beans_compl)

# Data adjustment
# All the effect columns must be as a factor to run in ASReml-r.
cols <- c("rep", "name", "loc", "mkt")
data_beans_compl[cols] <- lapply(data_beans_compl[cols], factor)
```

```
data_beans_compl <- data.table(data_beans_compl)
```

```
data_beans_compl
```

name	loc	mkt	rep	GY	DM	PH	LD
factor	factor	factor	factor	numeric	numeric	numeric	numeric
B1	BA	BB	1	2,481.7	89	16	1
B1	BA	BB	2	3,279.0	90	18	1
B1	BA	BB	3	2,479.2	90	14	1
B1	BA	BB	4	1,911.4	89	15	1
B2	BA	BB	1	3,159.8	90	20	2
B2	BA	BB	2	3,424.3	89	19	1
B2	BA	BB	3	2,655.0	91	15	2
B2	BA	BB	4	2,923.2	92	18	2
B4	BA	BB	1	2,565.5			
n: 1056							

Number of genotype per market class:

```
data_beans_compl_count <- data_beans_compl %>%
  means_by(name, mkt, na.rm = TRUE) %>%
  group_by(mkt) %>%
  dplyr::summarise(n = n())
```

```
data_beans_compl_count
```

mkt	n
factor	integer
BB	37
NB	37
SR	14

5.2.2 Box plot distribution

```
## Data visualization
```

```

#print(c)
a<- ggplot(data=data_beans_compl, aes(x=reorder(loc, -DM), y=DM, fill=loc)) +
  geom_boxplot()+
  facet_grid("mkt")+
  theme_bw()+
  theme(axis.text.x=element_text(angle = 90),
        strip.text=element_text(face="bold"), legend.position = "none")+
  labs(title="Days to Plant DM in days (DPM)",
        caption=NULL, x=NULL, y=NULL)

#print(a)

b<- ggplot(data=data_beans_compl, aes(x=reorder(loc, -PH), y=PH, fill=loc)) +
  geom_boxplot()+
  facet_grid("mkt")+
  theme_bw()+
  theme(axis.text.x=element_text(angle = 90),
        strip.text=element_text(face="bold"), legend.position = "none")+
  labs(title="Plant PH in cm (PH)",
        caption=NULL, x=NULL, y=NULL)

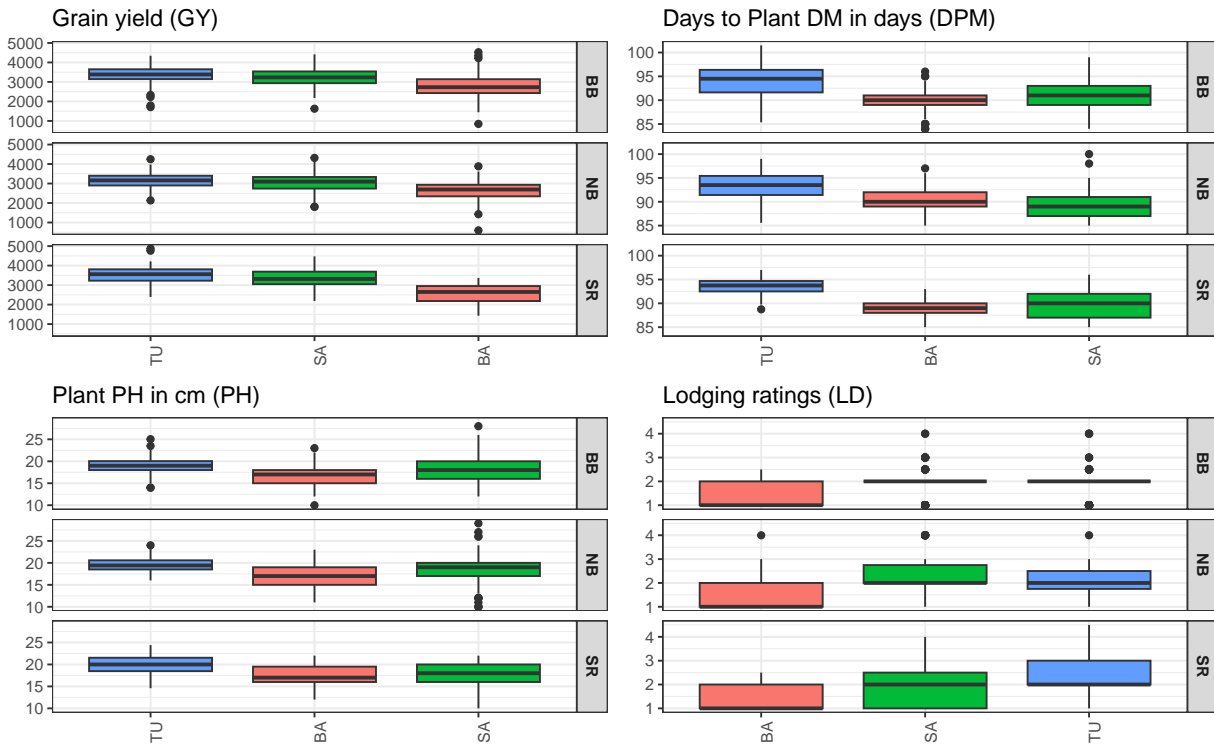
d<- ggplot(data=data_beans_compl, aes(x=reorder(loc, -LD), y=LD , fill=loc)) +
  geom_boxplot()+
  facet_grid("mkt")+
  theme_bw()+
  theme(axis.text.x=element_text(angle = 90),
        strip.text=element_text(face="bold"), legend.position = "none")+
  labs(title="Lodging ratings (LD)",
        caption=NULL, x=NULL, y=NULL)

c<- ggplot(data=data_beans_compl, aes(x=reorder(loc, -GY), y=GY , fill=loc)) +
  geom_boxplot()+
  facet_grid("mkt")+
  theme_bw()+
  theme(axis.text.x=element_text(angle = 90),
        strip.text=element_text(face="bold"), legend.position = "none")+
  labs(title="Grain yield (GY)",
        caption=NULL, x=NULL, y=NULL)

#print(a)

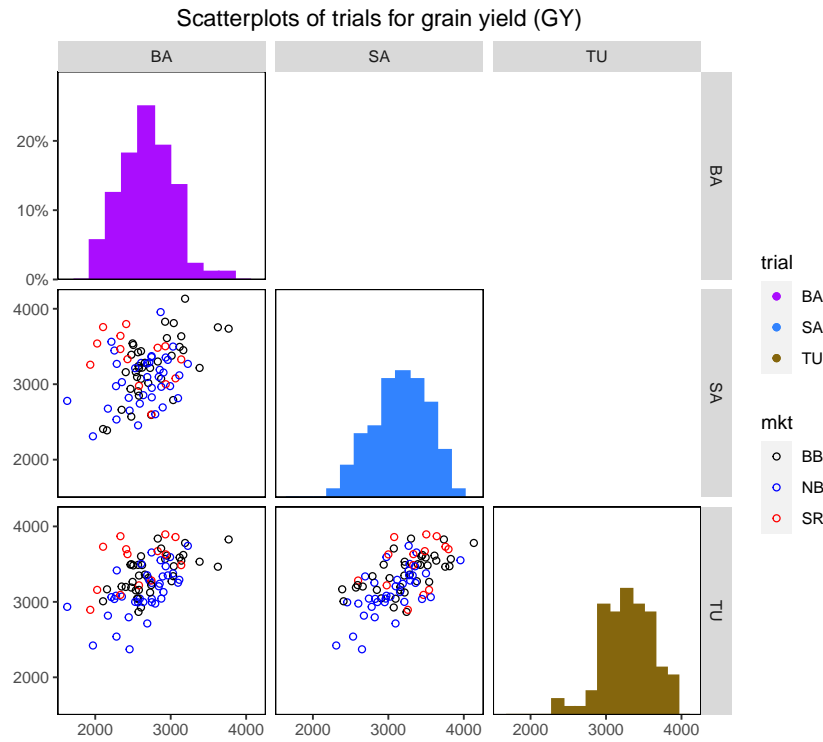
arrange_ggplot(c,a,b,d)

```



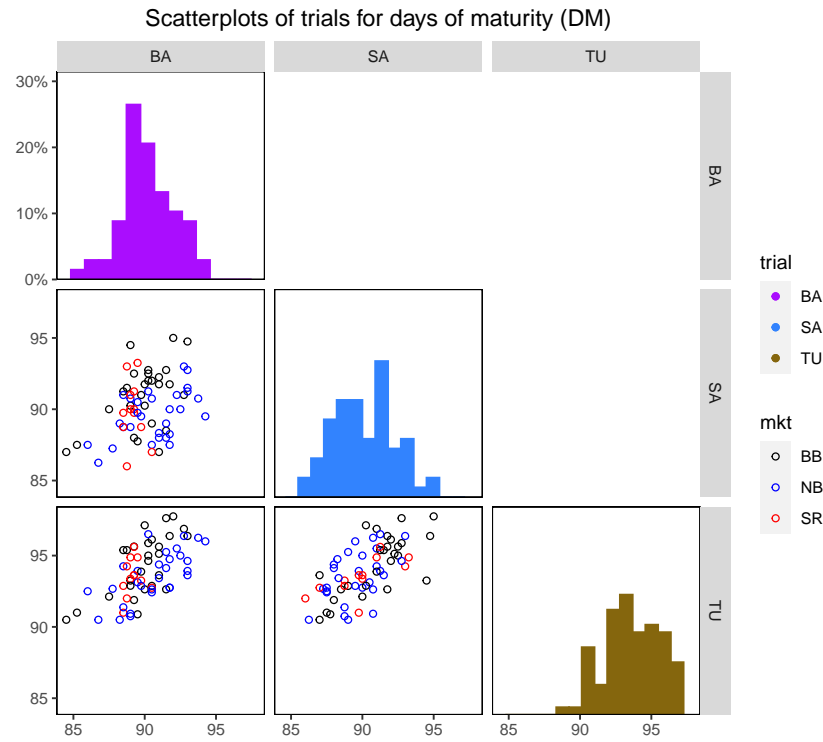
Box Plot for grain yield (GY), date of maturity (DM), Plant height (PH), and Lodging (LD) across environment

5.2.3 Scatter plot distribution GY



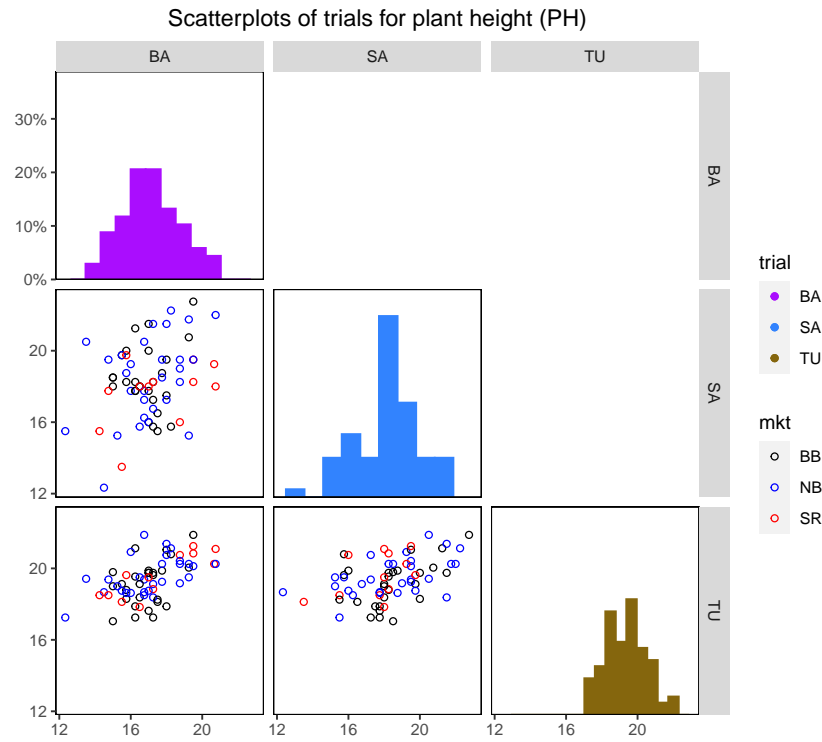
Scatterplot matrix of trials for grain yield (GY). The diagonal shows the distribution and the bottom left shows the scatter points for DM in each location.

5.2.4 Scatter plot distribution DM



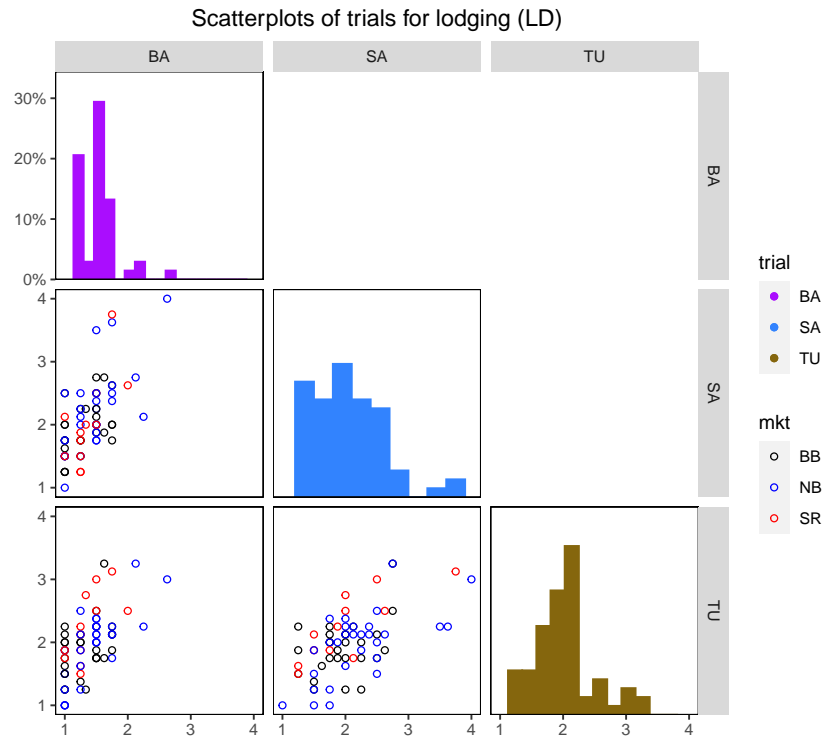
Scatterplot matrix of trials for days of maturity (DM). The diagonal shows the distribution and the bottom left shows the scatter points for DM in each location.

5.2.5 Scatter plot distribution PH



Scatterplot matrix of trials for plant height (PH). The diagonal shows the distribution and the bottom left shows the scatter points for PH in each location.

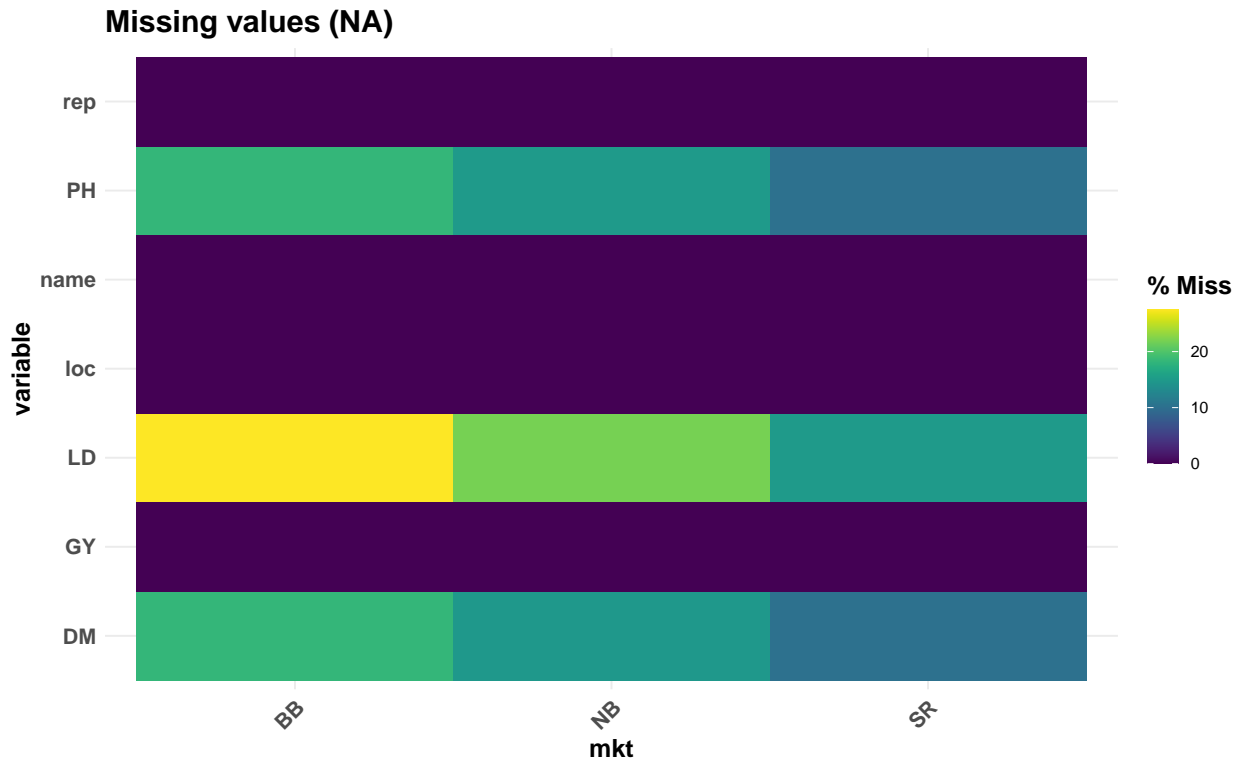
5.2.6 Scatter plot distribution LD



Scatterplot matrix of trials for lodging (LD). The diagonal shows the distribution and the bottom left shows the scatter points for LD in each location.

5.2.7 Missing values

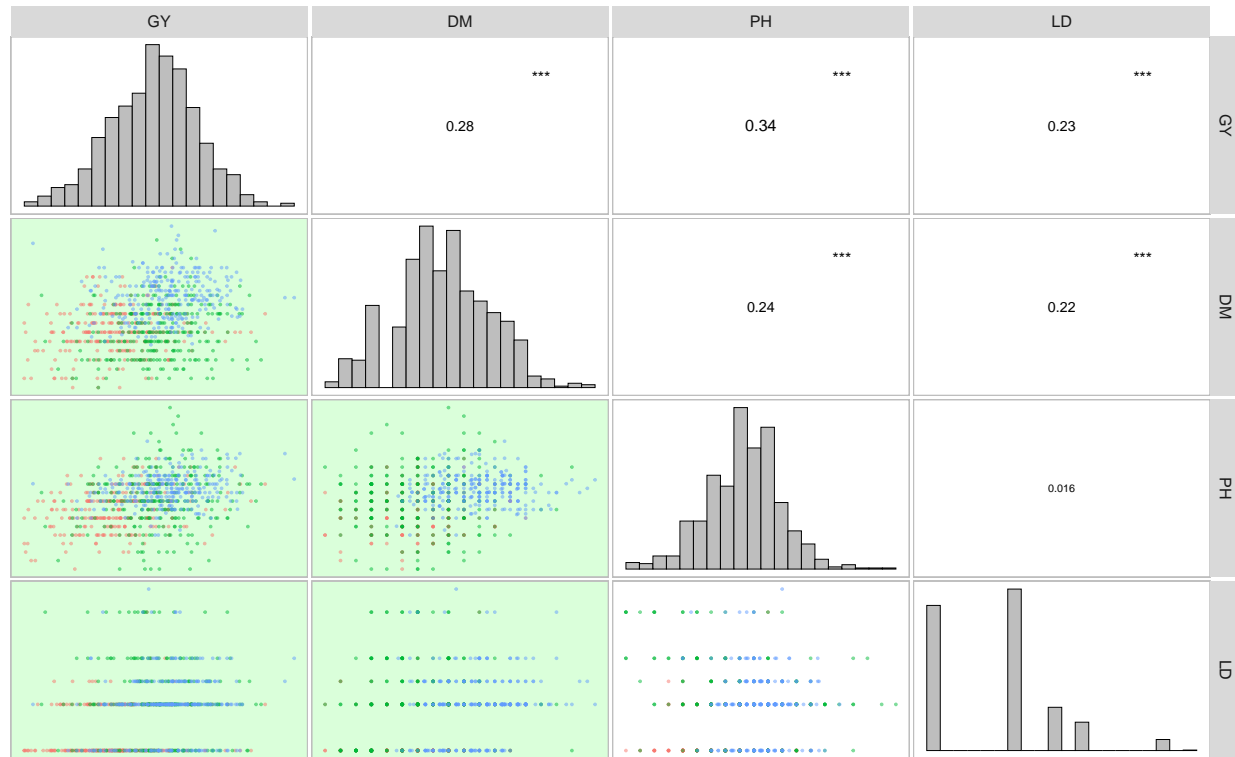
```
#str(data_beans_compl)
gg_miss_fct(data_beans_compl, fct = mkt) + ggtitle("Missing values (NA)") +
  theme(axis.text.x = element_text(face = "bold", size = 12),
        axis.text.y = element_text(face = "bold", size = 12),
        title = element_text(face = "bold", size = 14))
```



Missing genotypes by market classes

5.2.8 Linear phenotypic correlation

```
corr_plot(data_beans_compl,
  col.by = loc, # Different color for each level of ENV
  alpha.point = 0.5, # Color transparency of the points
  size.point = 0.3, # Size of the point
  size.line = 0.1) # The size of the line
```



Pearson's linear correlation between the studied traits. The lower diagonal shows the scatter plot where the environments are mapped with different point colors (BA = green, SA = salmon, TU = cyan)

5.3 Multi-trait stability index and selection gains

- For this study, the goal to response variables GY, DM, PH and LD are:
 - GY: higher (increase)
 - DM: smaller (decrease)
 - PH: higher (increase)
 - LD: smaller (decrease)
- The weights for the response variable are:
 - GY: 60 (60 yielding and 40 stability)
 - DM: 40 (40 maturity and 60 stability)
 - PH: 40 (40 height and 60 stability)
 - LD: 40 (40 lodging and 60 stability)
- Selection intensity (SI) of 20%

5.4 Black beans

```
data_beans_BB <- droplevels(subset(data_beans_compl, mkt == "BB"))

data_beans_BB <- droplevels(na.omit(data_beans_BB))
#str(data_beans_BB)
```

5.4.1 Multi-trait index selection

```

waasb_model_bb.1 <- waasb(data_beans_BB,
  env = loc,
  gen = name,
  rep = rep,
  resp = everything(),
  random = "gen", #Default
  verbose = F,
  wresp = c(60, 40, 40, 40),
  mresp = c("h,l,h,l") ) #'GY', 'DM', 'PH', LD
  #weight for response variable 60 and 40 for yielding and stability,
  ↪ respectively)

options(digits = 3)
mtsi1 = mtsi(waasb_model_bb.1, index = "waasby", SI = 20, mineval = 1, verbose = F)

Sel_waasb <- sel_gen(mtsi1)

mtsi1_resul<- mtsi1$sel_dif_trait

if (knitr::is_html_output()) {

  print_table(mtsi1_resul[,1:6])

}else{

mtsi1_resul[,1:6]
}

```

VAR	Factor	Xo	Xs	SD	SDperc
character	character	numeric	numeric	numeric	numeric
GY	FA 1	3,200.6	3,258.5	57.9	1.8
LD	FA 1	1.7	1.6	-0.1	-5.6
DM	FA 2	91.8	91.9	0.1	0.1
PH	FA 2	18.1	18.3	0.2	1.1

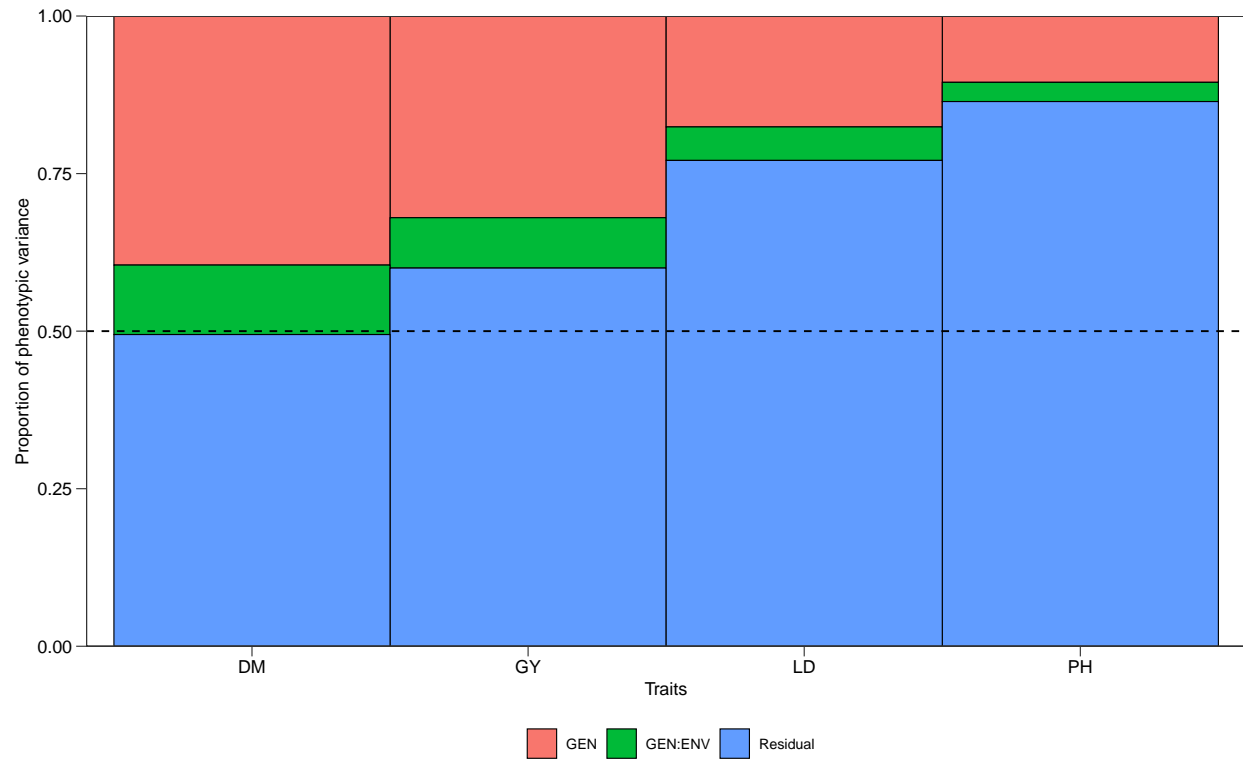
5.4.2 Model outputs

```

plot(waasb_model_bb.1,
  type = "vcomp", # Chose the type of plot.
  width.bar = 1, # No spaces between the bars
  size.line = 0.3) + # Controls the size of the line
  geom_hline(yintercept = 0.5, linetype = 2) # Add the dashed line

```

5.4.2.1 Variance plot



```
get_model_data(waasb_model_bb.1, "lrt")
```

5.4.2.2 Likelihood-ratio test

```
#> Class of the model: waasb
```

```
#> Variable extracted: lrt
```

VAR	model	npar	logLik	AIC	LRT	Df	Pr(>Chisq)
character	character	integer	numeric	numeric	numeric	numeric	numeric
GY	GEN	14	-2,333.5	4,695.1	25.1	1	0.0
GY	GEN:ENV	14	-2,322.9	4,673.9	4.0	1	0.0
DM	GEN	14	-720.5	1,469.0	30.2	1	0.0
DM	GEN:ENV	14	-710.2	1,448.5	9.7	1	0.0
PH	GEN	14	-733.1	1,494.3	6.1	1	0.0
PH	GEN:ENV	14	-730.3	1,488.5	0.4	1	0.5
LD	GEN	14	-289.8	607.5	12.3	1	0.0
LD	GEN:ENV	14	-284.3	596.6	1.3	1	0.2

VAR	model	npar	logLik	AIC	LRT	Df	Pr(>Chisq)
character	character	integer	numeric	numeric	numeric	numeric	numeric

```
get_model_data(waasb_model_bb.1, "vcomp")
```

5.4.2.3 Variance components

```
#> Class of the model: waasb
```

```
#> Variable extracted: vcomp
```

Group	GY	DM	PH	LD
character	numeric	numeric	numeric	numeric
GEN	68,571.6	2.9	0.6	0.1
GEN:ENV	17,115.1	0.8	0.2	0.0
Residual	128,582.1	3.6	5.2	0.3

```
get_model_data(waasb_model_bb.1, "genpar")
```

5.4.2.4 Genetic parameters

```
#> Class of the model: waasb
```

```
#> Variable extracted: genpar
```

Parameters	GY	DM	PH	LD
character	numeric	numeric	numeric	numeric
Phenotypic variance	214,268.9	7.2	6.0	0.4
Heritability	0.3	0.4	0.1	0.2
GEI _{r2}	0.1	0.1	0.0	0.1
h ² _{mg}	0.8	0.8	0.6	0.7
Accuracy	0.9	0.9	0.7	0.8
r _{ge}	0.1	0.2	0.0	0.1
CV _g	8.2	1.8	4.4	14.5

Parameters	GY	DM	PH	LD
character	numeric	numeric	numeric	numeric
CVr	11.2	2.1	12.5	30.3
CV ratio	0.7	0.9	0.3	0.5

```
get_model_data(waasb_model_bb.1, "details")
```

5.4.2.5 Models details

```
#> Class of the model: waasb
```

```
#> Variable extracted: details
```

Parameters	GY	DM	PH	LD
character	character	character	character	character
Mean	3199.25	91.77	18.14	1.73
SE	30.37	0.19	0.15	0.04
SD	544.12	3.32	2.67	0.67
CV	17.03	3.63	14.75	38.72
Min	1601.88 (B43 in BA)	84 (B22 in BA)	10 (B27 in BA)	1 (B1 in BA)
Max	4524.05 (B26 in BA)	101.5 (B61 in TU)	28 (B40 in SA)	4 (B25 in SA)
MinENV	BA (2828.83)	BA (89.91)	BA (16.86)	BA (1.32)
MaxENV	TU (3405.26)	TU (94.44)	TU (19.13)	SA (1.96)
MinGEN	B45 (2570.2)	B45 (87.33)	B60 (16.85)	B14 (1.25)
n: 14				

```
get_model_data(waasb_model_bb.1, "fixed")
```

5.4.2.6 Fixed effects

```
#> Class of the model: waasb
```

```
#> Variable extracted: fixed
```

VAR	SOURCE	Sum Sq	Mean Sq	NumDF	DenDF	F value	Pr(>F)
character	character	numeric	numeric	integer	numeric	numeric	numeric
GY	ENV	3,275,190.1	1,637,595.1	2	243.1	12.7	0.0
GY	ENV:REP	6,661,855.0	740,206.1	9	229.8	5.8	0.0
DM	ENV	303.7	151.9	2	221.8	42.4	0.0
DM	ENV:REP	110.2	12.2	9	232.4	3.4	0.0
PH	ENV	47.2	23.6	2	267.7	4.6	0.0
PH	ENV:REP	152.1	16.9	9	233.3	3.3	0.0
LD	ENV	7.1	3.6	2	260.5	13.0	0.0
LD	ENV:REP	6.1	0.7	9	233.1	2.5	0.0

```
get_model_data(waasb_model_bb.1, "blupge") %>%
  means_by(ENV)
```

5.4.2.7 Enviroment means

```
#> Class of the model: waasb
```

```
#> Variable extracted: blupge
```

ENV	GY	DM	PH	LD
factor	numeric	numeric	numeric	numeric
BA	2,835.2	89.9	16.9	1.3
SA	3,362.1	91.0	18.4	2.0
TU	3,401.9	94.4	19.1	1.9

5.4.3 Selection Description

5.4.3.1 Genotype ranking

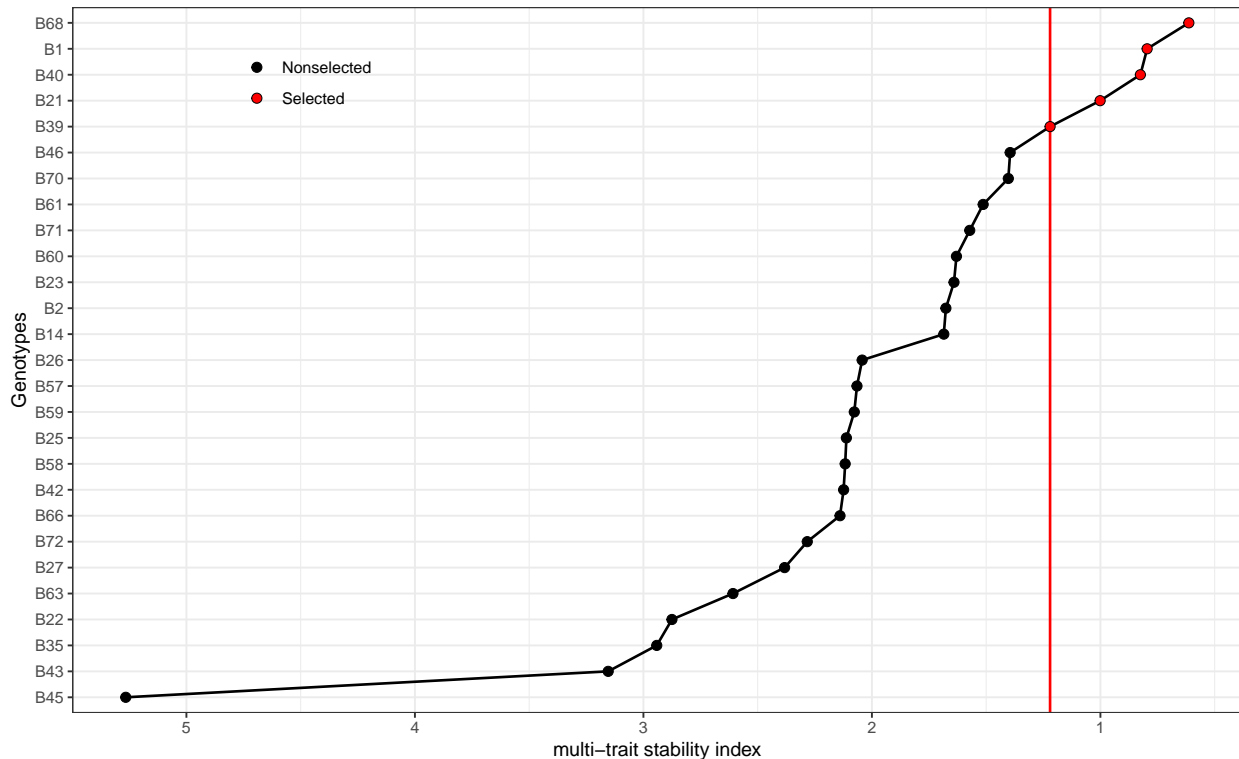
- Varieties ranking based on the multi-trait stability index. Selected varieties are highlighted in red.

```
# Get the random effects
#get_model_data(waasb_model_bb.1, what = "ranef")

mtsi1_value<- mtsi1$MTSI

#plot(mtsi1, arrange.label = TRUE)
```

```
p2 = plot(mtsi1, SI = 20, radar = FALSE) +
  coord_flip() +
  theme_bw() +
  labs(x = "Genotypes", y = "multi-trait stability index") +
  theme(legend.position = c(0.2, 0.9),
        legend.background = element_blank(),
        legend.key = element_blank(),
        legend.title = element_blank())
print(p2)
```



Cultivars ranking based on the multi-trait stability index. Selected cultivars are highlighted in red.

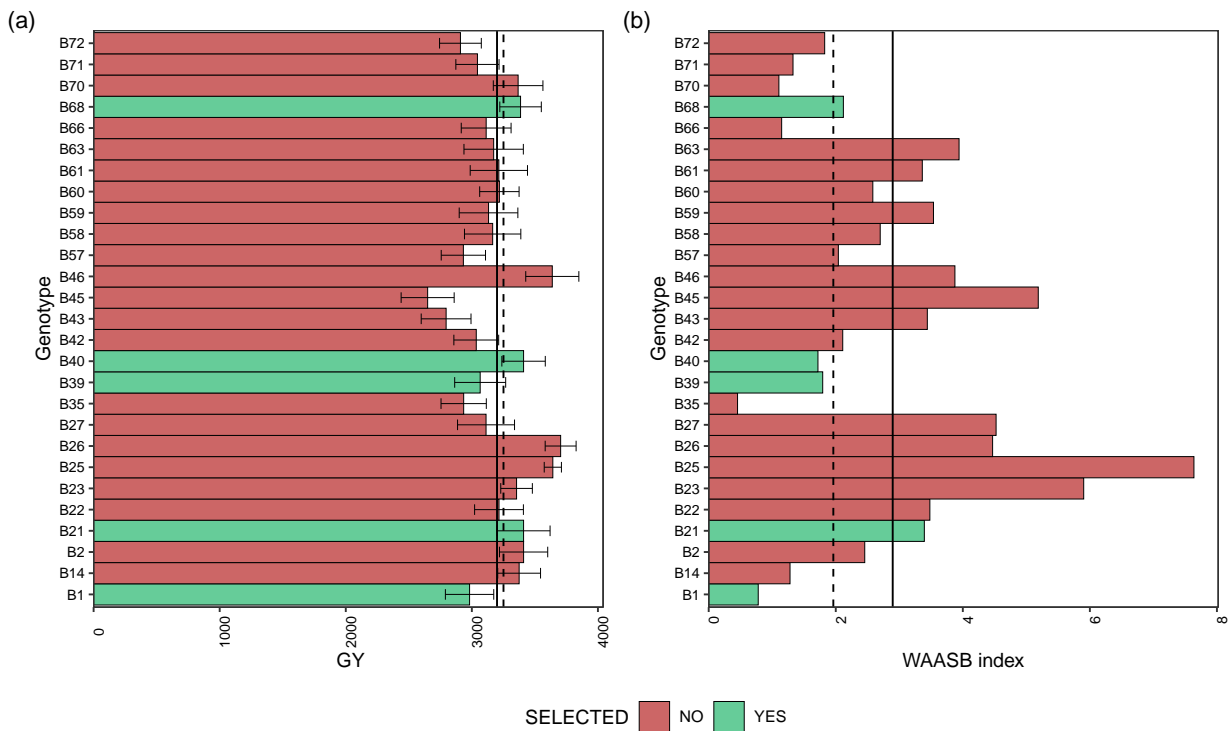
```
Sel_waasb
```

```
#> [1] "B68" "B1" "B40" "B21" "B39"
```

```
p1<- plot_selected(blups_sel, GEN, GY, blups_sel_mean_GY$mean_GY)
p2<- plot_selected(waasb_sel, GEN, GY, waasb_sel_mean_GY$mean_GY) + labs(y = "WAASB
  ↪ index")

arrange_ggplot(p1, p2,
  guides = "collect",
  tag_levels = "a",
```

```
tag_prefix = "(",
tag_suffix = ")")
```

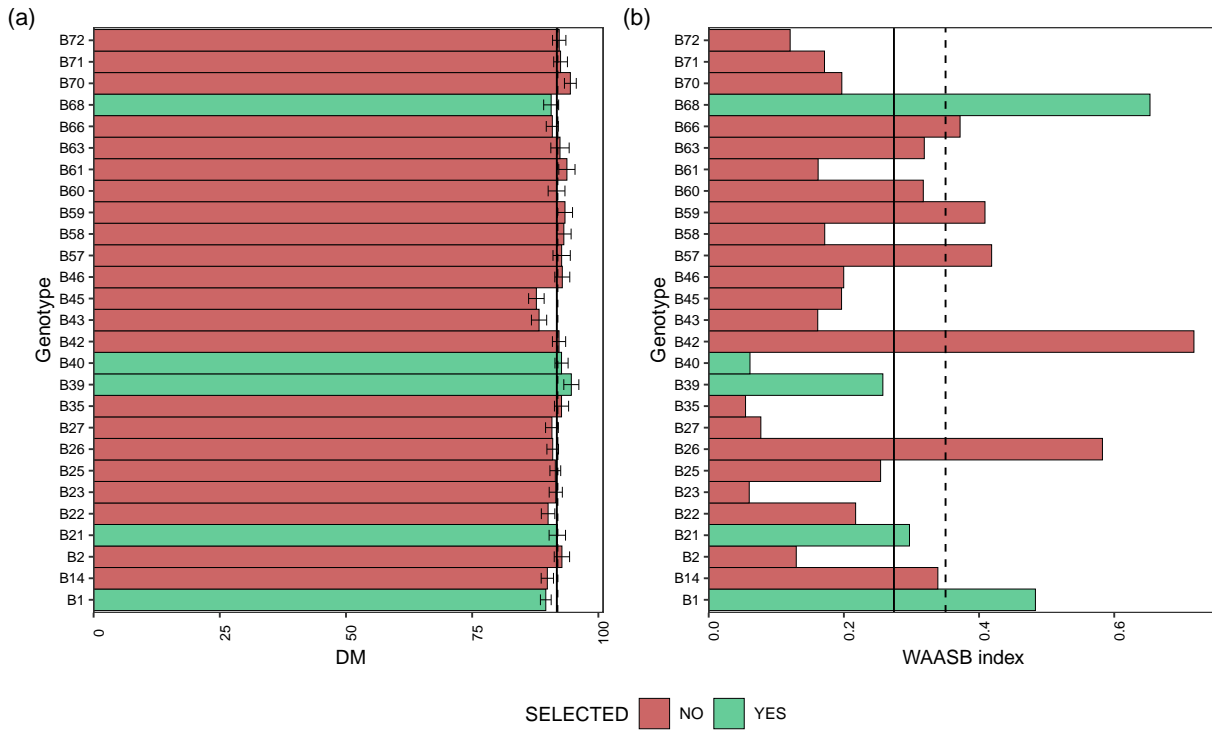


Mean performance (a) and stability (b) for grain yield (GY) of 27 black beans genotypes. The vertical dashed and solid lines shows, respectively, the mean of the selected genotype and the overall mean for both mean performance and WAASB index

5.4.3.2 Mean performance and stability for GY

```
p1<- plot_selected(blups_sel, GEN, DM, blups_sel_mean_DM$mean_DM)
p2<- plot_selected(waasb_sel, GEN, DM, waasb_sel_mean_DM$mean_DM) + labs(y = "WAASB
  ↪ index")

arrange_ggplot(p1, p2,
  guides = "collect",
  tag_levels = "a",
  tag_prefix = "(",
  tag_suffix = ")")
```

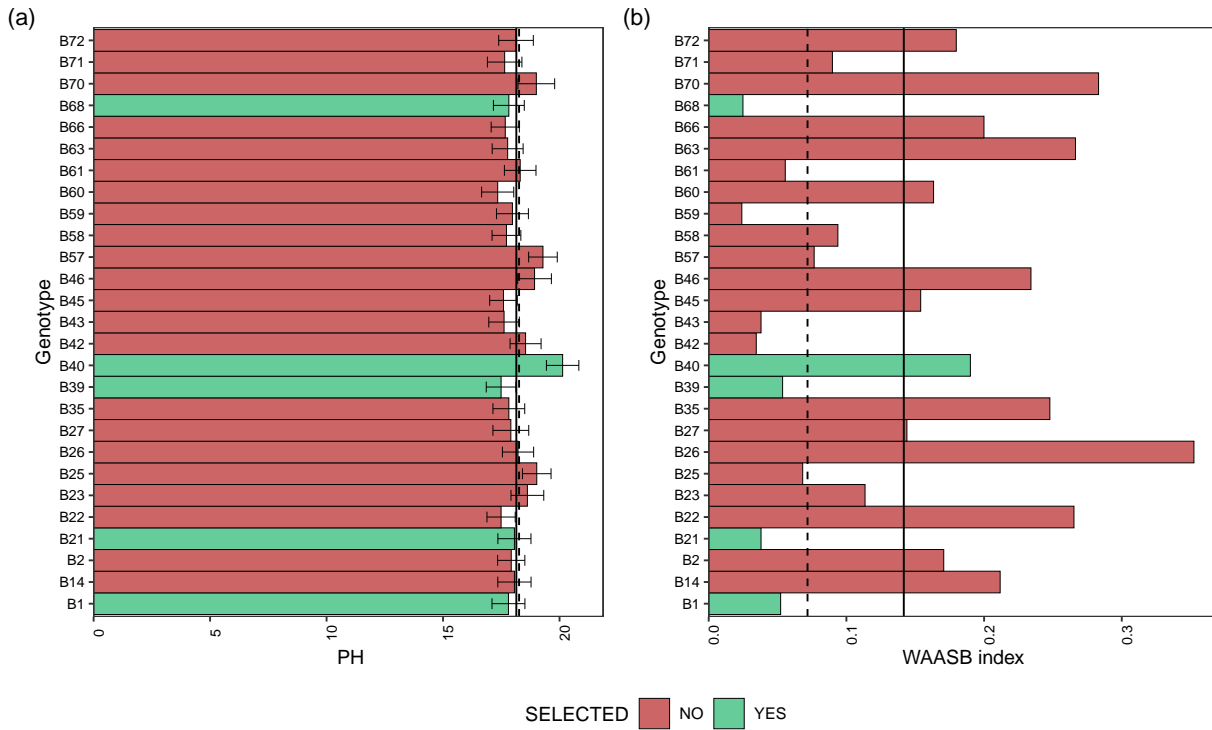


Mean performance (a) and stability (b) for days of maturity (DM) of 27 black beans genotypes. The vertical dashed and solid lines shows, respectively, the mean of the selected genotype and the overall mean for both mean performance and WAASB index

5.4.3.3 Mean performance and stability for DM

```
p1<- plot_selected(blups_sel, GEN, PH, blups_sel_mean_PH$mean_PH)
p2<- plot_selected(waasb_sel, GEN, PH, waasb_sel_mean_PH$mean_PH) + labs(y = "WAASB
  ↪ index")

arrange_ggplot(p1, p2,
  guides = "collect",
  tag_levels = "a",
  tag_prefix = "(",
  tag_suffix = ")")
```

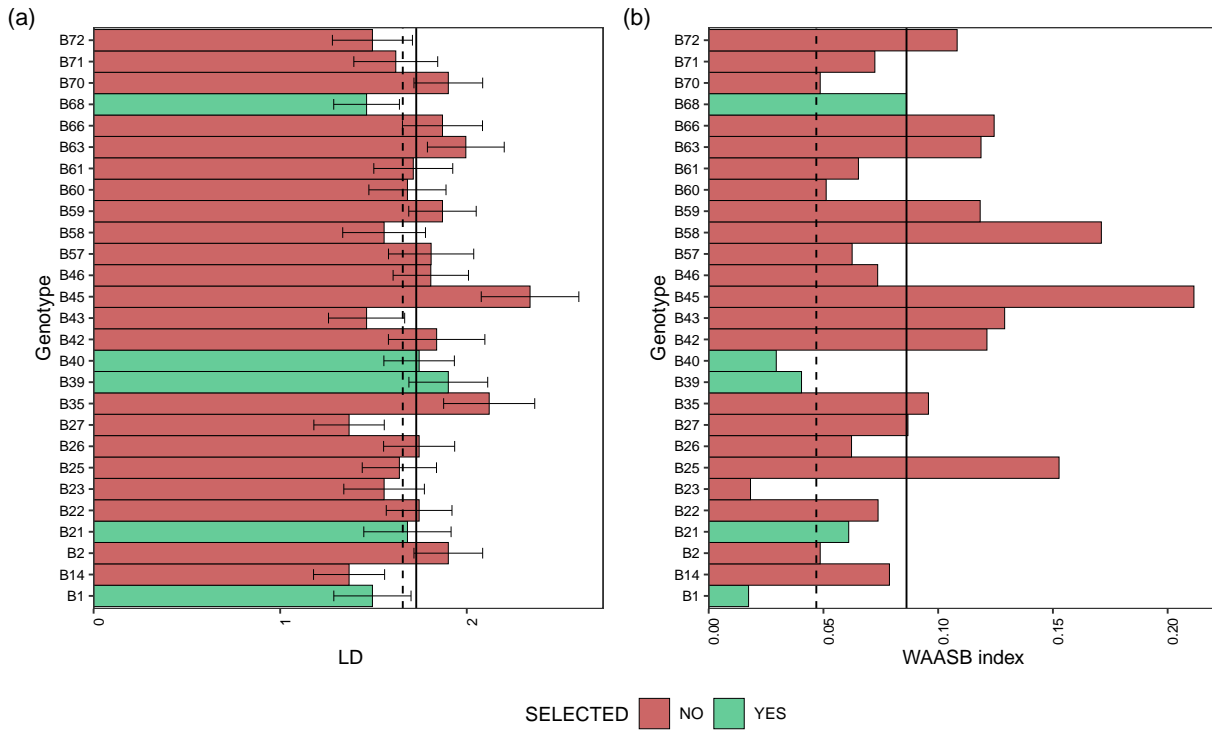


Mean performance (a) and stability (b) for plant height (PH) of 27 black beans genotypes. The vertical dashed and solid lines shows, respectively, the mean of the selected genotype and the overall mean for both mean performance and WAASB index

5.4.3.4 Mean performance and stability for PH

```
p1<- plot_selected(blups_sel, GEN, LD, blups_sel_mean_LD$mean_LD)
p2<- plot_selected(waasb_sel, GEN, LD, waasb_sel_mean_LD$mean_LD) + labs(y = "WAASB
  ↪ index")

arrange_ggplot(p1, p2,
  guides = "collect",
  tag_levels = "a",
  tag_prefix = "(",
  tag_suffix = ")")
```



Mean performance (a) and stability (b) for lodging (LD) of 27 black beans genotypes. The vertical dashed and solid lines shows, respectively, the mean of the selected genotype and the overall mean for both mean performance and WAASB index

5.4.3.5 Mean performance and stability for LD

5.5 Navy beans

```
data_beans_NB <- droplevels(subset(data_beans_compl, mkt == "NB"))
data_beans_NB <- droplevels(na.omit(data_beans_NB))
#str(data_beans_NB)
```

5.5.1 Multi-trait index selection

```
waasb_model_nb.1 <- waasb(data_beans_NB,
  env = loc,
  gen = name,
  rep = rep,
  resp = everything(),
  random = "gen", #Default
  verbose = F,
  wresp = c(60, 40, 40, 40),
```



```

mresp = c("h, l,h, l") ) #'GY', 'DM', 'PH', LD
#weight for response variable 60 and 40 for yielding and stability,
↪ respectively)

options(digits = 3)
mtsi1 = mtsi(waasb_model_nb.1, index = "waasby", SI = 20, mineval = 1, verbose = F)

mtsi1_resul<- mtsi1$sel_dif_trait

Sel_waasb <- sel_gen(mtsi1)

if (knitr::is_html_output()) {

  print_table(mtsi1_resul[,1:6])

}else{

mtsi1_resul[,1:6]
}

```

VAR	Factor	Xo	Xs	SD	SDperc
character	character	numeric	numeric	numeric	numeric
GY	FA 1	3,006.1	3,145.0	138.9	4.6
DM	FA 1	91.3	91.5	0.3	0.3
PH	FA 1	18.3	19.7	1.4	7.4
LD	FA 1	1.9	1.7	-0.2	-11.9

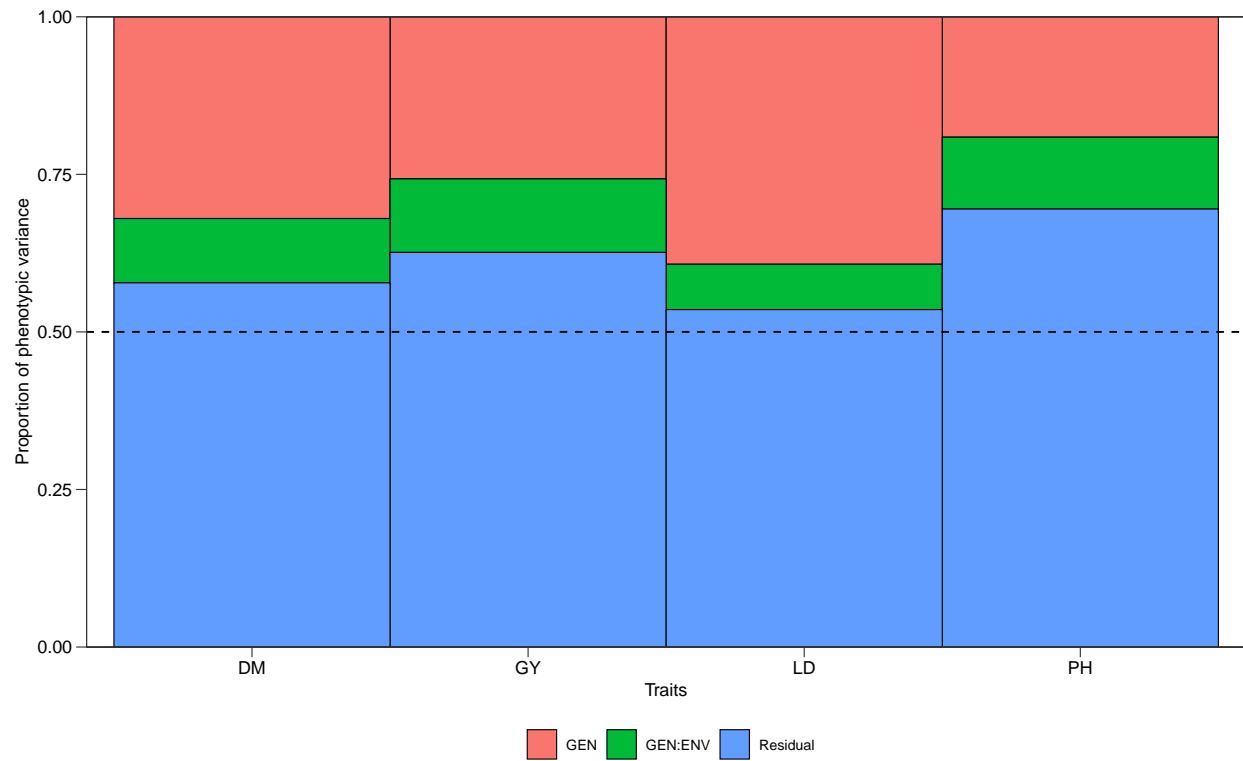
5.5.2 Model outputs

```

plot(waasb_model_nb.1,
type = "vcomp", # Chose the type of plot.
width.bar = 1, # No spaces between the bars
size.line = 0.3) + # Controls the size of the line
geom_hline(yintercept = 0.5, linetype = 2) # Add the dashed line

```

5.5.2.1 Variance plot



```
get_model_data(waasb_model_nb.1, "lrt")
```

5.5.2.2 Likelihood-ratio test

```
#> Class of the model: waasb
```

```
#> Variable extracted: lrt
```

VAR	model	npar	logLik	AIC	LRT	Df	Pr(>Chisq)
character	character	integer	numeric	numeric	numeric	numeric	numeric
GY	GEN	14	-2,501.2	5,030.4	17.9	1	0.0
GY	GEN:ENV	14	-2,496.2	5,020.3	7.8	1	0.0
DM	GEN	14	-763.5	1,555.0	25.4	1	0.0
DM	GEN:ENV	14	-754.4	1,536.8	7.2	1	0.0
PH	GEN	14	-803.6	1,635.2	12.1	1	0.0
PH	GEN:ENV	14	-800.6	1,629.3	6.1	1	0.0
LD	GEN	14	-335.4	698.9	36.0	1	0.0
LD	GEN:ENV	14	-319.7	667.4	4.5	1	0.0

```
get_model_data(waasb_model_nb.1, "vcomp")
```

5.5.2.3 Variance components

```
#> Class of the model: waasb
```

```
#> Variable extracted: vcomp
```

Group	GY	DM	PH	LD
character	numeric	numeric	numeric	numeric
GEN	50,195.5	2.0	1.4	0.2
GEN:ENV	22,779.4	0.6	0.8	0.0
Residual	122,206.2	3.6	4.9	0.3

```
get_model_data(waasb_model_nb.1, "genpar")
```

5.5.2.4 Genetic parameters

```
#> Class of the model: waasb
```

```
#> Variable extracted: genpar
```

Parameters	GY	DM	PH	LD
character	numeric	numeric	numeric	numeric
Phenotypic variance	195,181.1	6.2	7.1	0.5
Heritability	0.3	0.3	0.2	0.4
GEI _{r2}	0.1	0.1	0.1	0.1
h ² _{mg}	0.7	0.8	0.7	0.9
Accuracy	0.9	0.9	0.8	0.9
r _{ge}	0.2	0.2	0.1	0.1
CV _g	7.4	1.5	6.3	23.6
CV _r	11.6	2.1	12.1	27.6
CV ratio	0.6	0.7	0.5	0.9

```
get_model_data(waasb_model_nb.1, "details")
```

5.5.2.5 Models details

```
#> Class of the model: waasb
```

```
#> Variable extracted: details
```

Parameters	GY	DM	PH	LD
character	character	character	character	character
Mean	3009.36	91.28	18.37	1.87
SE	26.04	0.16	0.15	0.04
SD	483.58	3.02	2.84	0.77
CV	16.09	3.31	15.46	41.04
Min	1618.91 (N35 in BA)	85 (N15 in BA)	10 (N38 in SA)	1 (N3 in BA)
Max	4313.9 (N36 in SA)	100 (N57 in SA)	29 (N23 in SA)	4 (N47 in BA)
MinENV	BA (2737.08)	SA (89.57)	BA (17.05)	BA (1.46)
MaxENV	TU (3195.31)	TU (93.57)	TU (19.57)	SA (2.19)
MinGEN	N15 (2272.31)	N15 (87.82)	N15 (15.27)	N15 (1)
n: 14				

```
get_model_data(waasb_model_nb.1, "fixed")
```

5.5.2.6 Fixed effects

```
#> Class of the model: waasb
```

```
#> Variable extracted: fixed
```

VAR	SOURCE	Sum Sq	Mean Sq	NumDF	DenDF	F value	Pr(>F)
character	character	numeric	numeric	integer	numeric	numeric	numeric
GY	ENV	2,987,371.0	1,493,685.5	2	247.7	12.2	0.0
GY	ENV:REP	2,887,992.3	320,888.0	9	250.1	2.6	0.0
DM	ENV	240.1	120.0	2	251.3	33.7	0.0

VAR	SOURCE	Sum Sq	Mean Sq	NumDF	DenDF	F value	Pr(>F)
character	character	numeric	numeric	integer	numeric	numeric	numeric
DM	ENV:REP	114.4	12.7	9	250.3	3.6	0.0
PH	ENV	82.4	41.2	2	254.2	8.3	0.0
PH	ENV:REP	66.4	7.4	9	247.6	1.5	0.2
LD	ENV	4.7	2.4	2	262.2	8.9	0.0
LD	ENV:REP	8.3	0.9	9	249.5	3.5	0.0

```
get_model_data(waasb_model_nb.1, "blupge") %>%
  means_by(ENV)
```

5.5.2.7 Enviroment means

```
#> Class of the model: waasb
```

```
#> Variable extracted: blupge
```

ENV	GY	DM	PH	LD
factor	numeric	numeric	numeric	numeric
BA	2,732.0	90.6	17.0	1.5
SA	3,094.1	89.6	18.5	2.2
TU	3,195.3	93.6	19.6	2.0

5.5.3 Selection Description

5.5.3.1 Genotype ranking

- Varieties ranking based on the multi-trait stability index. Selected varieties are highlighted in red.

```
# Get the random effects
#get_model_data(waasb_model_nb.1, what = "ranef")

mtsi1_value<- mtsi1$MTSI

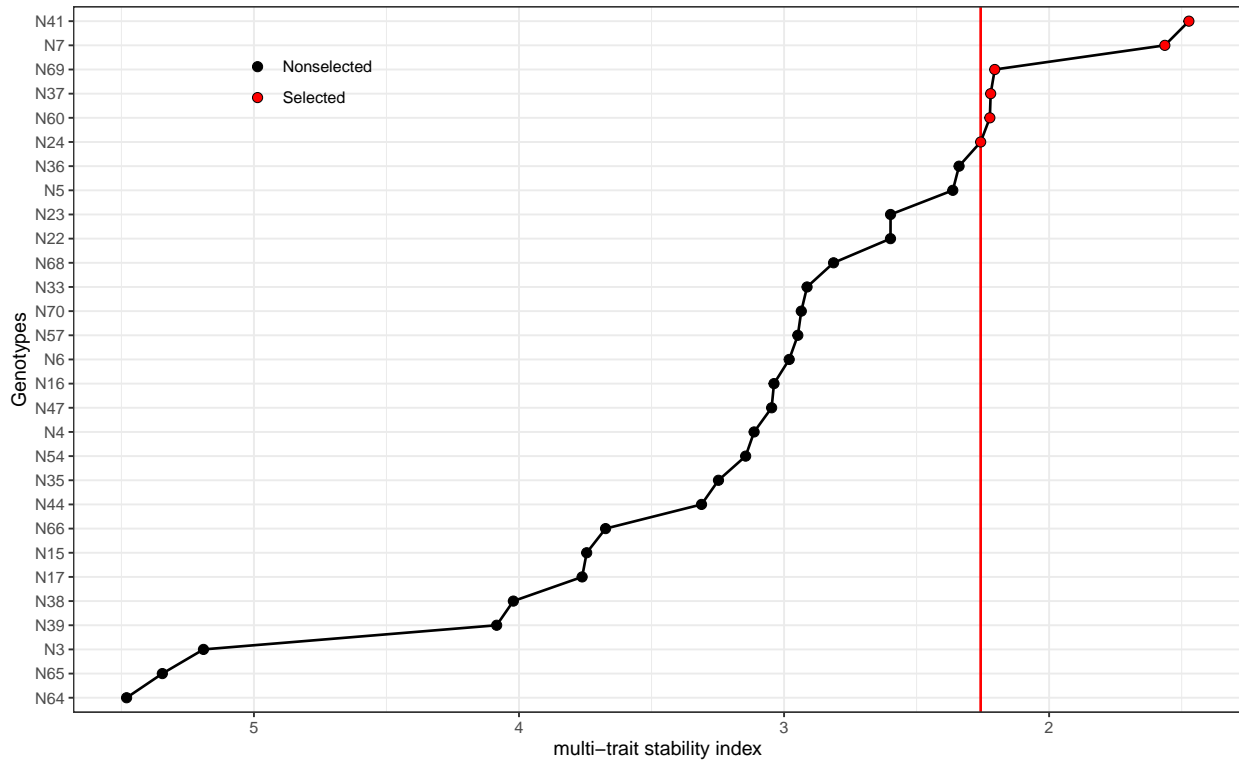
#plot(mtsi1, arrange.label = TRUE)

p2 = plot(mtsi1, SI = 20, radar = FALSE) +
  coord_flip() +
  theme_bw() +
  labs(x = "Genotypes", y = "multi-trait stability index") +
  theme(legend.position = c(0.2, 0.9),
```

```

legend.background = element_blank(),
legend.key = element_blank(),
legend.title = element_blank()
print(p2)

```



Cultivars ranking based on the multi-trait stability index. Selected cultivars are highlighted in red.

```
Sel_waasb
```

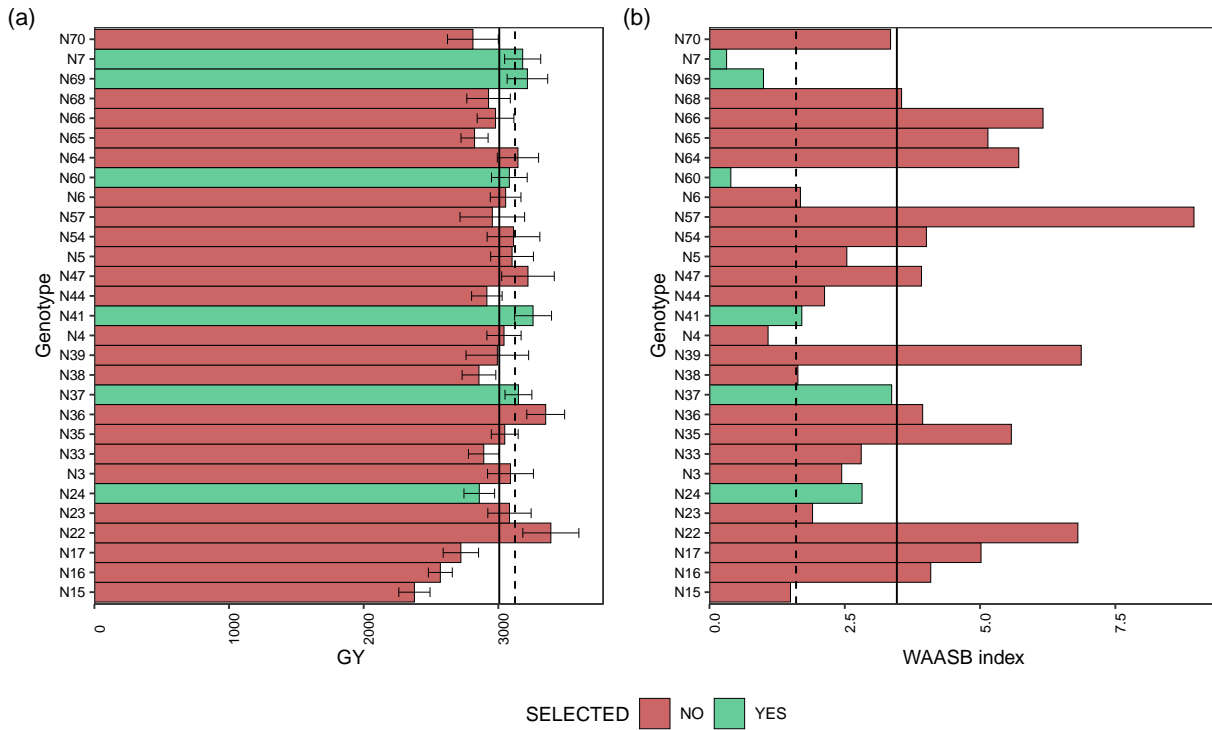
```
#> [1] "N41" "N7" "N69" "N37" "N60" "N24"
```

```

p1<- plot_selected(blups_sel, GEN, GY, blups_sel_mean_GY$mean_GY)
p2<- plot_selected(waasb_sel, GEN, GY, waasb_sel_mean_GY$mean_GY) + labs(y = "WAASB
  ↪ index")

arrange_ggplot(p1, p2,
  guides = "collect",
  tag_levels = "a",
  tag_prefix = "(",
  tag_suffix = ")")

```

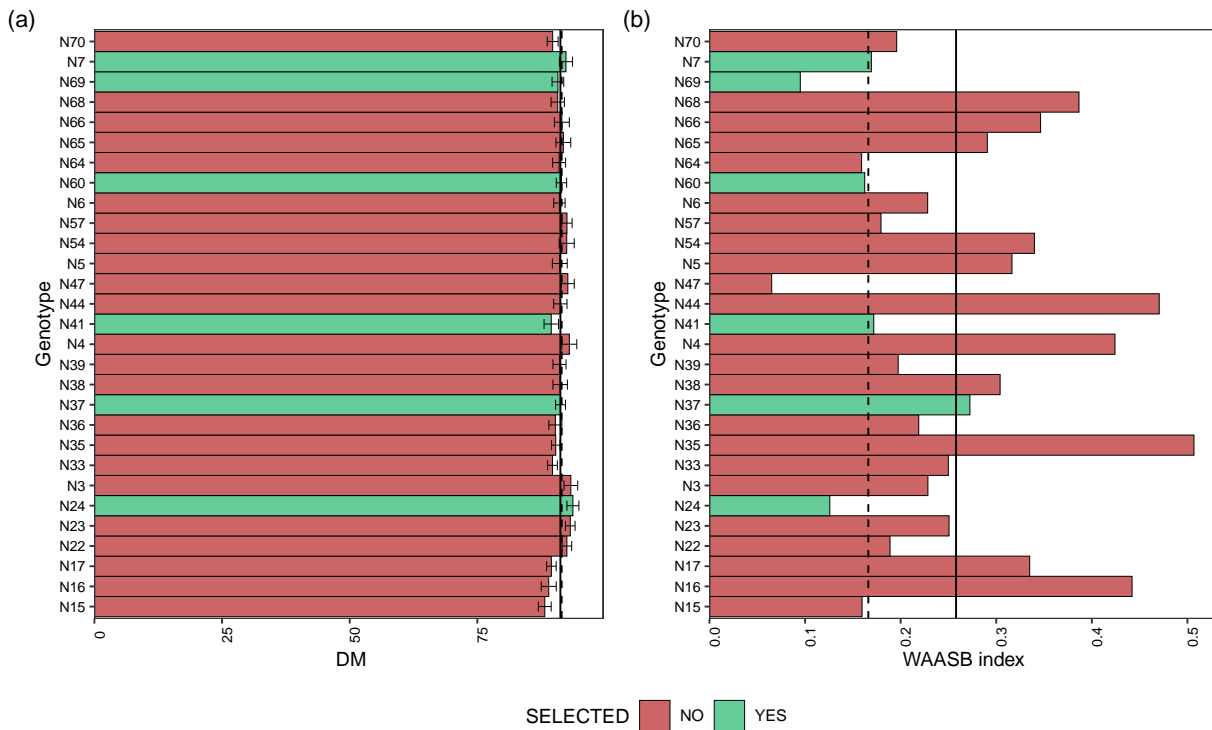


Mean performance (a) and stability (b) for grain yield (GY) of 29 navy beans genotypes. The vertical dashed and solid lines shows, respectively, the mean of the selected genotype and the overall mean for both mean performance and WAASB index

5.5.3.2 Mean performance and stability for GY

```
p1<- plot_selected(blups_sel, GEN, DM, blups_sel_mean_DM$mean_DM)
p2<- plot_selected(waasb_sel, GEN, DM, waasb_sel_mean_DM$mean_DM) + labs(y = "WAASB
  ↪ index")

arrange_ggplot(p1, p2,
  guides = "collect",
  tag_levels = "a",
  tag_prefix = "(",
  tag_suffix = ")")
```

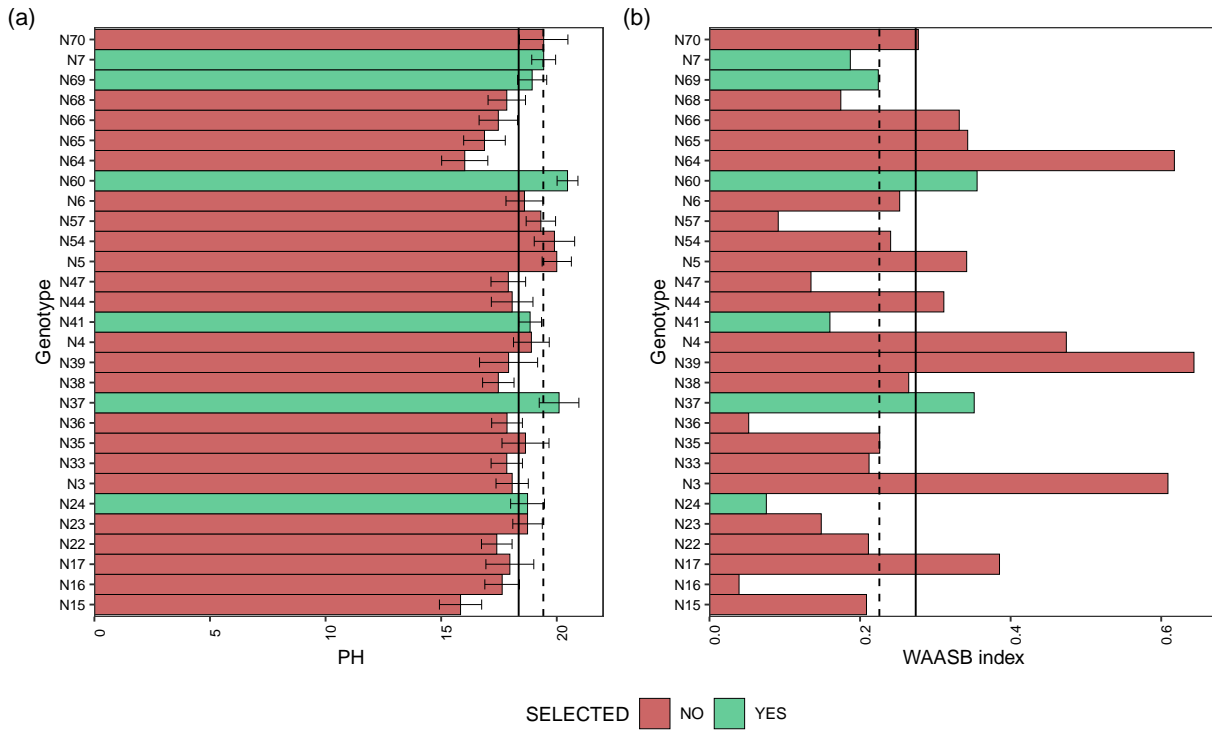


Mean performance (a) and stability (b) for days of maturity (DM) of 29 navy beans genotypes. The vertical dashed and solid lines shows, respectively, the mean of the selected genotype and the overall mean for both mean performance and WAASB index

5.5.3.3 Mean performance and stability for DM

```
p1<- plot_selected(blups_sel, GEN, PH, blups_sel_mean_PH$mean_PH)
p2<- plot_selected(waasb_sel, GEN, PH, waasb_sel_mean_PH$mean_PH) + labs(y = "WAASB
  ↪ index")

arrange_ggplot(p1, p2,
  guides = "collect",
  tag_levels = "a",
  tag_prefix = "(",
  tag_suffix = ")")
```

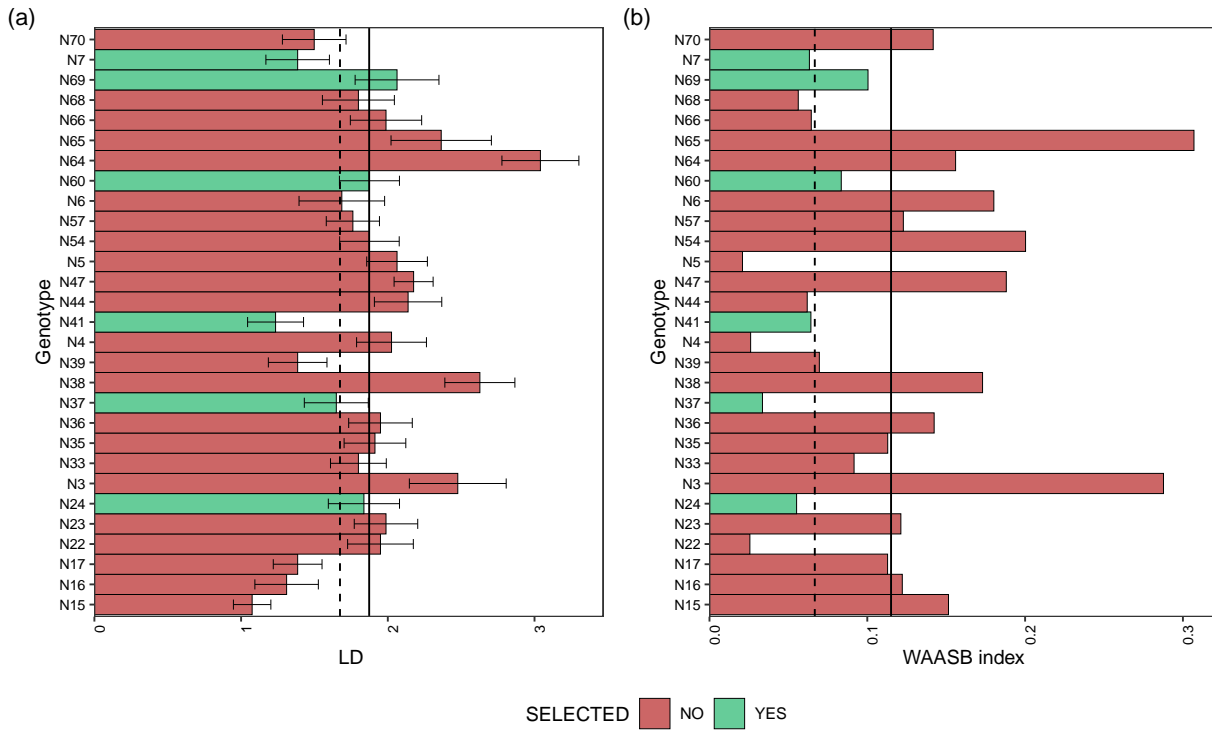



Mean performance (a) and stability (b) for plant height (PH) of 29 navy beans genotypes. The vertical dashed and solid lines shows, respectively, the mean of the selected genotype and the overall mean for both mean performance and WAASB index

5.5.3.4 Mean performance and stability for PH

```
p1<- plot_selected(blups_sel, GEN, LD, blups_sel_mean_LD$mean_LD)
p2<- plot_selected(waasb_sel, GEN, LD, waasb_sel_mean_LD$mean_LD) + labs(y = "WAASB
  ↪ index")

arrange_ggplot(p1, p2,
  guides = "collect",
  tag_levels = "a",
  tag_prefix = "(",
  tag_suffix = ")")
```



Mean performance (a) and stability (b) for lodging (LD) of 29 navy beans genotypes. The vertical dashed and solid lines shows, respectively, the mean of the selected genotype and the overall mean for both mean performance and WAASB index

5.5.3.5 Mean performance and stability for LD

5.6 Red beans

```
data_beans_SR <- droplevels(subset(data_beans_compl, mkt == "SR"))

data_beans_SR <- droplevels(na.omit(data_beans_SR))
#str(data_beans_SR)
```

5.6.1 Multi-trait index selection

```
waasb_model_sr.1 <- waasb(data_beans_SR,
  env = loc,
  gen = name,
  rep = rep,
  resp = everything(),
  random = "gen", #Default
  verbose = F,
  wresp = c(60, 40, 40, 40),
  mresp = c("h, l, h, l") ) #'GY', 'DM', 'PH', LD
```

```

#weight for response variable 60 and 40 for yielding and stability,
↪ respectively)

options(digits = 3)
mtsi1 = mtsi(waasb_model_sr.1, index = "waasby", SI = 20, mineval = 1, verbose = F)

mtsi1_resul<- mtsi1$sel_dif_trait

Sel_waasb <- sel_gen(mtsi1)

if (knitr::is_html_output()) {
  print_table(mtsi1_resul[,1:6])
}else{
mtsi1_resul[,1:6]
}

```

VAR	Factor	Xo	Xs	SD	SDperc
character	character	numeric	numeric	numeric	numeric
PH	FA 1	18.3	20.1	1.8	10.0
LD	FA 1	1.9	1.9	0.0	2.1
GY	FA 2	3,155.9	3,161.8	5.9	0.2
DM	FA 2	90.8	90.9	0.0	0.0

```
#print_table(mtsi1_resul[,1:6])
```

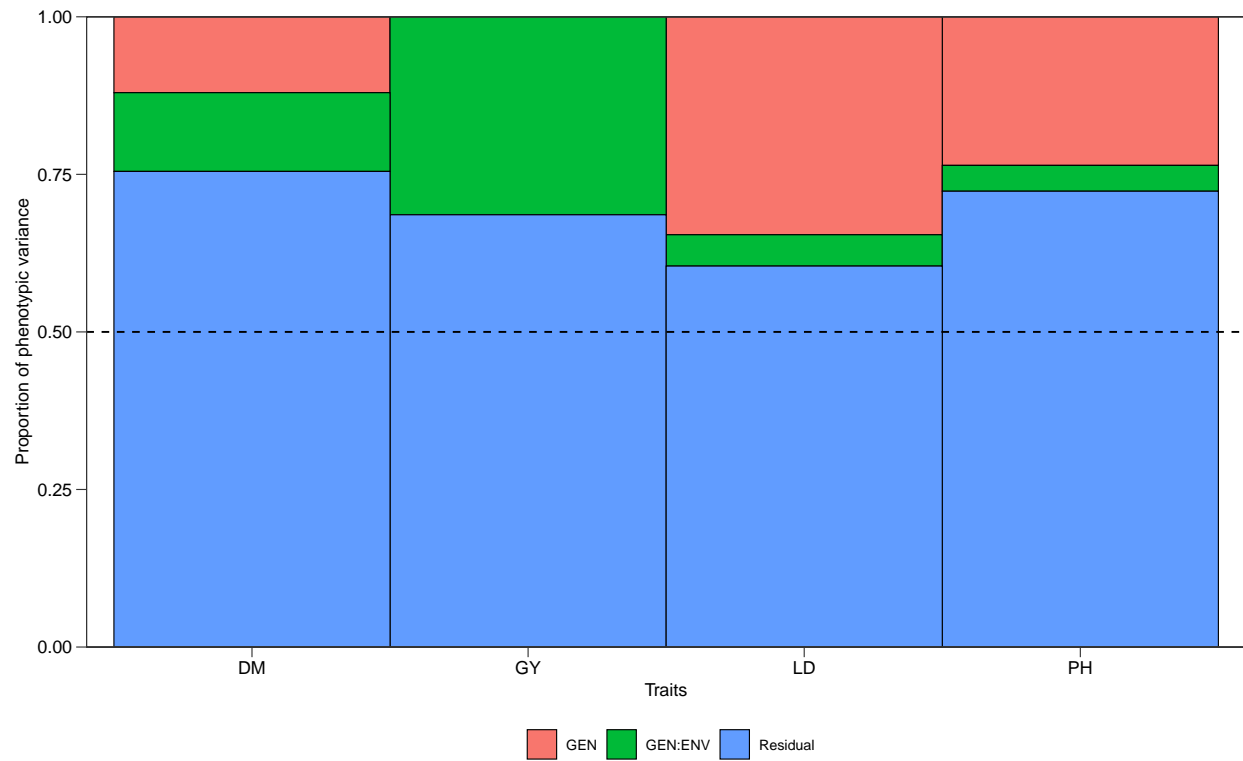
5.6.2 Model outputs

```

plot(waasb_model_sr.1,
type = "vcomp", # Chose the type of plot.
width.bar = 1, # No spaces between the bars
size.line = 0.3) + # Controls the size of the line
geom_hline(yintercept = 0.5, linetype = 2) # Add the dashed line

```

5.6.2.1 Variance plot



```
get_model_data(waasb_model_sr.1, "lrt")
```

5.6.2.2 Likelihood-ratio test

```
#> Class of the model: waasb
```

```
#> Variable extracted: lrt
```

VAR	model	npar	logLik	AIC	LRT	Df	Pr(>Chisq)
character	character	integer	numeric	numeric	numeric	numeric	numeric
GY	GEN	14	-1,005.1	2,038.1	-0.0	1	1.0
GY	GEN:ENV	14	-1,011.2	2,050.3	12.2	1	0.0
DM	GEN	14	-302.1	632.2	2.3	1	0.1
DM	GEN:ENV	14	-302.3	632.5	2.6	1	0.1
PH	GEN	14	-323.0	674.0	8.1	1	0.0
PH	GEN:ENV	14	-319.1	666.2	0.4	1	0.5
LD	GEN	14	-147.4	322.7	13.0	1	0.0
LD	GEN:ENV	14	-141.3	310.5	0.8	1	0.4

```
get_model_data(waasb_model_sr.1, "vcomp")
```

5.6.2.3 Variance components

```
#> Class of the model: waasb
```

```
#> Variable extracted: vcomp
```

Group	GY	DM	PH	LD
character	numeric	numeric	numeric	numeric
GEN	0.0	0.6	1.7	0.2
GEN:ENV	75,950.2	0.6	0.3	0.0
Residual	165,821.6	3.8	5.1	0.3

```
get_model_data(waasb_model_sr.1, "genpar")
```

5.6.2.4 Genetic parameters

```
#> Class of the model: waasb
```

```
#> Variable extracted: genpar
```

Parameters	GY	DM	PH	LD
character	numeric	numeric	numeric	numeric
Phenotypic variance	241,771.8	5.1	7.1	0.5
Heritability	0.0	0.1	0.2	0.3
GEI _{r2}	0.3	0.1	0.0	0.0
h ² _{mg}	0.0	0.5	0.8	0.8
Accuracy	0.0	0.7	0.9	0.9
r _{ge}	0.3	0.1	0.1	0.1
CV _g	0.0	0.9	7.1	22.9
CV _r	12.9	2.1	12.4	30.2
CV ratio	0.0	0.4	0.6	0.8

```
get_model_data(waasb_model_sr.1, "details")
```

5.6.2.5 Models details

```
#> Class of the model: waasb
```

```
#> Variable extracted: details
```

Parameters	GY	DM	PH	LD
character	character	character	character	character
Mean	3157.34	90.86	18.23	1.87
SE	53.6	0.25	0.24	0.07
SD	638.78	2.95	2.8	0.85
CV	20.3	3.26	15.42	45.39
Min	1643.85 (R10 in BA)	85 (R5 in BA)	10 (R5 in SA)	1 (R2 in BA)
Max	4876.34 (R10 in TU)	97 (R9 in TU)	23.59 (R12 in TU)	4.5 (R16 in TU)
MinENV	BA (2579)	BA (89.17)	BA (17.45)	BA (1.36)
MaxENV	TU (3546.13)	TU (93.51)	TU (19.59)	TU (2.24)
MinGEN	R15 (2875.55)	R13 (88.92)	R16 (15.71)	R12 (1.33)
n: 14				

```
get_model_data(waasb_model_sr.1, "fixed")
```

5.6.2.6 Fixed effects

```
#> Class of the model: waasb
```

```
#> Variable extracted: fixed
```

VAR	SOURCE	Sum Sq	Mean Sq	NumDF	DenDF	F value	Pr(>F)
character	character	numeric	numeric	integer	numeric	numeric	numeric
GY	ENV	3,444,789.9	1,722,394.9	2	101.3	10.4	0.0
GY	ENV:REP	2,181,509.9	242,390.0	9	98.2	1.5	0.2
DM	ENV	174.4	87.2	2	99.8	22.9	0.0

VAR	SOURCE	Sum Sq	Mean Sq	NumDF	DenDF	F value	Pr(>F)
character	character	numeric	numeric	integer	numeric	numeric	numeric
DM	ENV:REP	68.4	7.6	9	98.5	2.0	0.0
PH	ENV	26.0	13.0	2	111.3	2.5	0.1
PH	ENV:REP	61.0	6.8	9	97.9	1.3	0.2
LD	ENV	2.5	1.2	2	108.6	3.9	0.0
LD	ENV:REP	13.4	1.5	9	98.4	4.6	0.0

```
get_model_data(waasb_model_sr.1, "blupge") %>%
  means_by(ENV)
```

5.6.2.7 Enviroment means

```
#> Class of the model: waasb
```

```
#> Variable extracted: blupge
```

ENV	GY	DM	PH	LD
factor	numeric	numeric	numeric	numeric
BA	2,583.6	89.2	17.5	1.4
SA	3,334.8	89.9	17.6	2.0
TU	3,546.1	93.5	19.6	2.2

5.6.3 Selection Description

5.6.3.1 Genotype ranking

- Varieties ranking based on the multi-trait stability index. Selected varieties are highlighted in red.

```
# Get the random effects
#get_model_data(waasb_model_sr.1, what = "ranef")

mtsi1_value<- mtsi1$MTSI

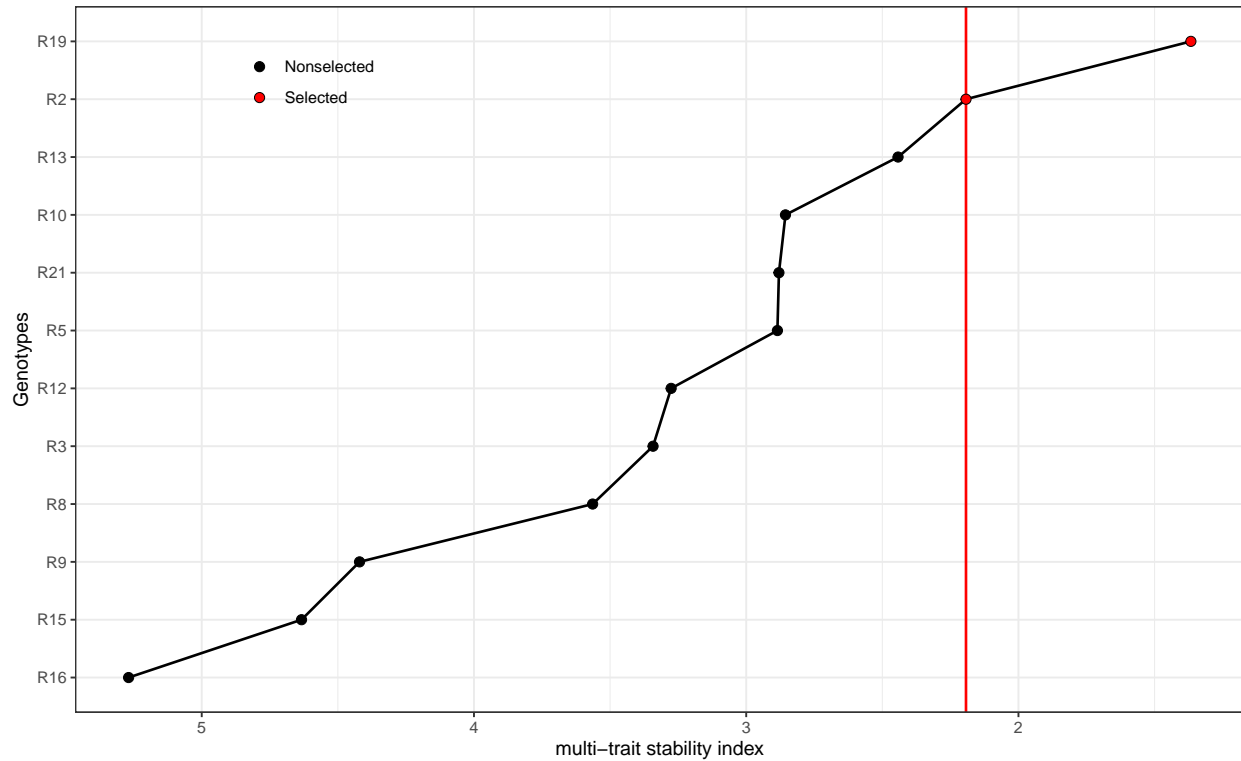
#plot(mtsi1, arrange.label = TRUE)

p2 = plot(mtsi1, SI = 20, radar = FALSE) +
  coord_flip() +
  theme_bw() +
  labs(x = "Genotypes", y = "multi-trait stability index") +
  theme(legend.position = c(0.2, 0.9),
```

```

legend.background = element_blank(),
legend.key = element_blank(),
legend.title = element_blank()
print(p2)

```



Cultivars ranking based on the multi-trait stability index. Selected cultivars are highlighted in red.

```
Sel_waasb
```

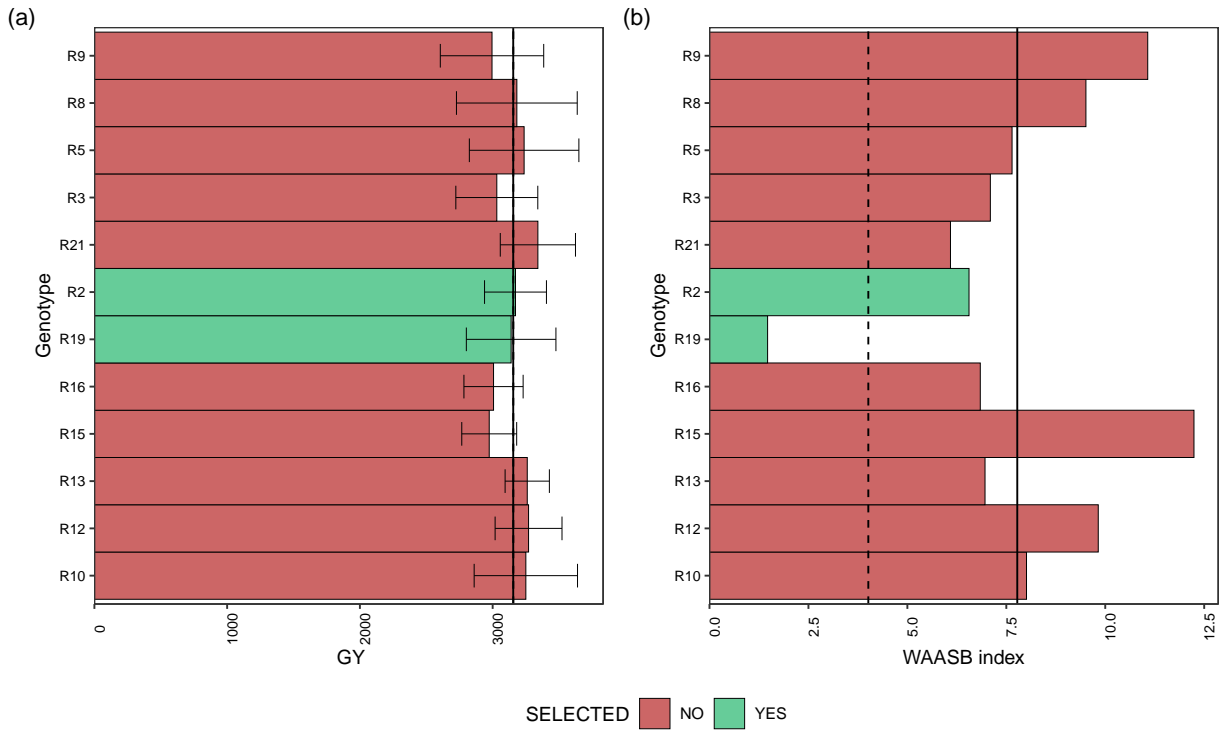
```
#> [1] "R19" "R2"
```

```

p1<- plot_selected(blups_sel, GEN, GY, blups_sel_mean_GY$mean_GY)
p2<- plot_selected(waasb_sel, GEN, GY, waasb_sel_mean_GY$mean_GY) + labs(y = "WAASB
  ↪ index")

arrange_ggplot(p1, p2,
  guides = "collect",
  tag_levels = "a",
  tag_prefix = "(",
  tag_suffix = ")")

```

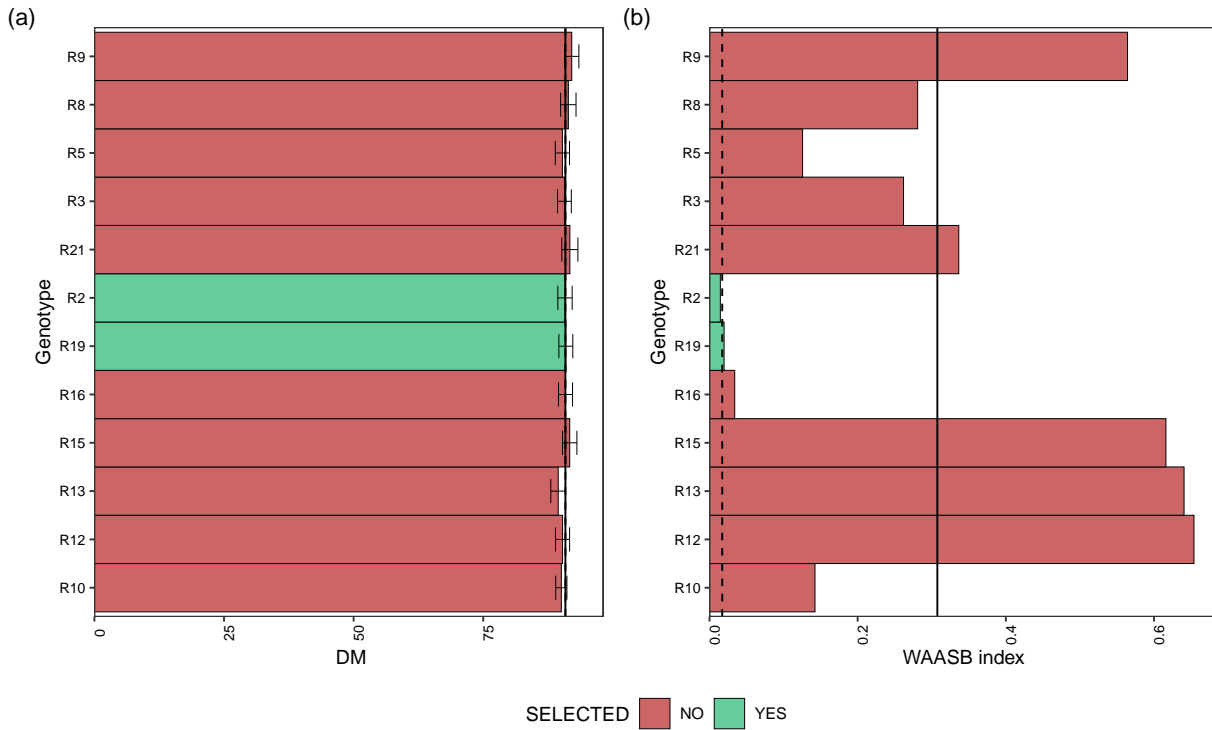



Mean performance (a) and stability (b) for grain yield (GY) of 12 red beans genotypes. The vertical dashed and solid lines shows, respectively, the mean of the selected genotype and the overall mean for both mean performance and WAASB index

5.6.3.2 Mean performance and stability for GY

```
p1<- plot_selected(blups_sel, GEN, DM, blups_sel_mean_DM$mean_DM)
p2<- plot_selected(waasb_sel, GEN, DM, waasb_sel_mean_DM$mean_DM) + labs(y = "WAASB
  ↪ index")

arrange_ggplot(p1, p2,
  guides = "collect",
  tag_levels = "a",
  tag_prefix = "(",
  tag_suffix = ")")
```

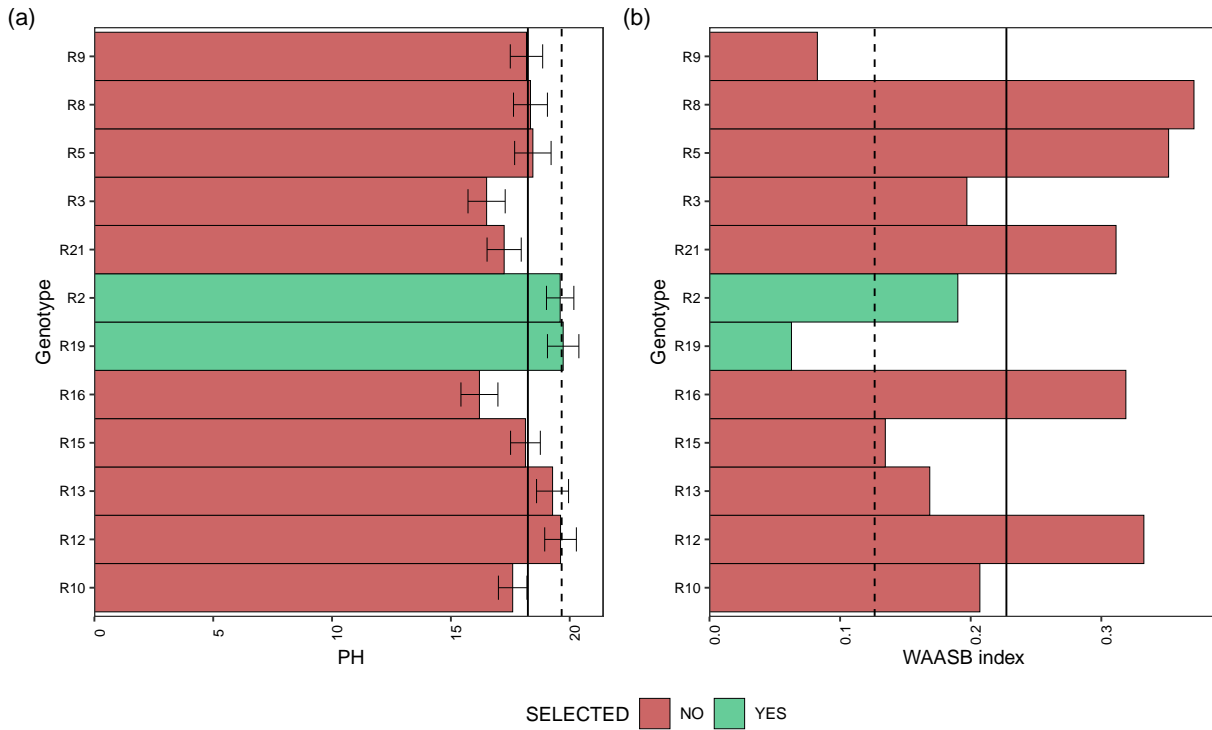


Mean performance (a) and stability (b) for days of maturity (DM) of 12 red beans genotypes. The vertical dashed and solid lines shows, respectively, the mean of the selected genotype and the overall mean for both mean performance and WAASB index

5.6.3.3 Mean performance and stability for DM

```
p1<- plot_selected(blups_sel, GEN, PH, blups_sel_mean_PH$mean_PH)
p2<- plot_selected(waasb_sel, GEN, PH, waasb_sel_mean_PH$mean_PH) + labs(y = "WAASB
  ↪ index")

arrange_ggplot(p1, p2,
  guides = "collect",
  tag_levels = "a",
  tag_prefix = "(",
  tag_suffix = ")")
```

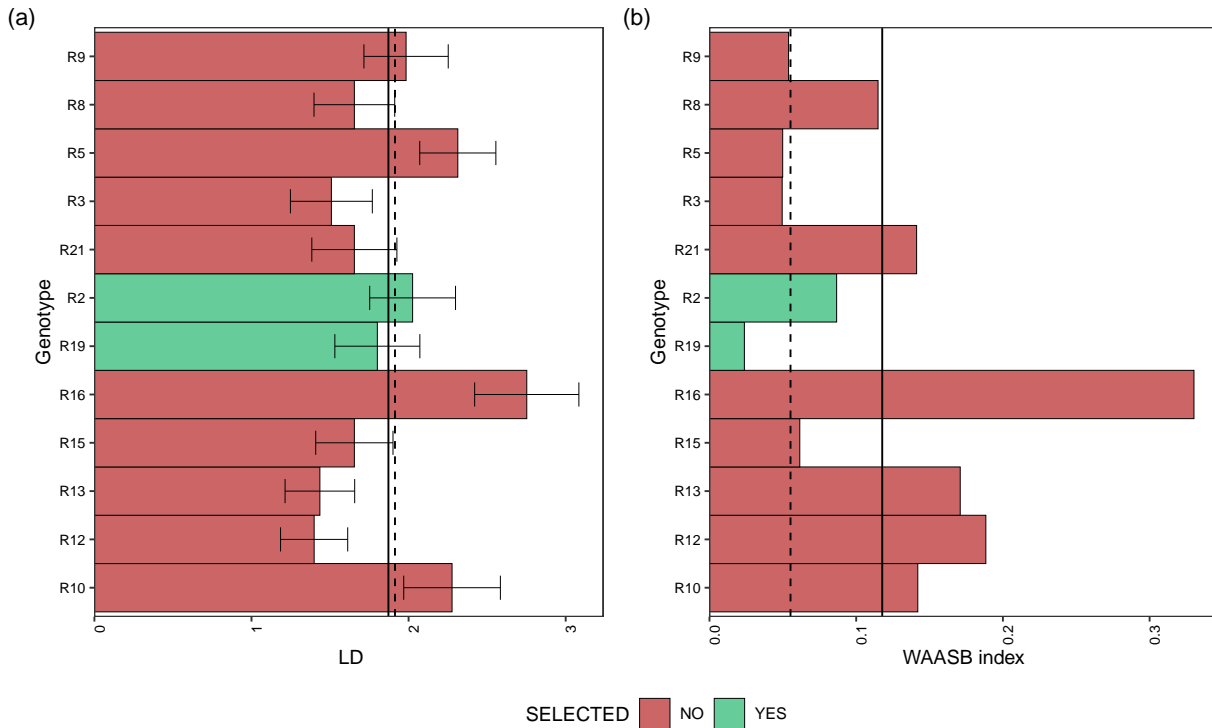


Mean performance (a) and stability (b) for plant height (PH) of 12 red beans genotypes. The vertical dashed and solid lines shows, respectively, the mean of the selected genotype and the overall mean for both mean performance and WAASB index

5.6.3.4 Mean performance and stability for PH

```
p1<- plot_selected(blups_sel, GEN, LD, blups_sel_mean_LD$mean_LD)
p2<- plot_selected(waasb_sel, GEN, LD, waasb_sel_mean_LD$mean_LD) + labs(y = "WAASB
  ↪ index")

arrange_ggplot(p1, p2,
  guides = "collect",
  tag_levels = "a",
  tag_prefix = "(",
  tag_suffix = ")")
```



Mean performance (a) and stability (b) for lodging (LD) of 12 red beans genotypes. The vertical dashed and solid lines shows, respectively, the mean of the selected genotype and the overall mean for both mean performance and WAASB index

5.6.3.5 Mean performance and stability for LD

5.7 MTME mixed model

5.7.1 Black beans

```
data_beans_BB <- droplevels(data_beans_compl[which(data_beans_compl$mkt=="BB")])
#TU
data_beans_BB.sc.TU <- droplevels(subset(data_beans_BB, loc=="TU"))
data_beans_BB.sc.TU <- scale(data_beans_BB.sc.TU[,5:8])
#dim(data_beans_BB.sc.TU)
#BA
data_beans_BB.sc.BA <- droplevels(subset(data_beans_BB, loc=="BA"))
data_beans_BB.sc.BA <- scale(data_beans_BB.sc.BA[,5:8])

#SA
data_beans_BB.sc.SA <- droplevels(subset(data_beans_BB, loc=="SA"))
data_beans_BB.sc.SA <- scale(data_beans_BB.sc.SA[,5:8])

#data_beans_BB.sc<- scale(data_beans_BB[,5:6])
data_beans_BB<- data_beans_BB %>%
  dplyr::select(name, loc, mkt,rep)
```

```

data_beans_BB.sc<- rbind(data_beans_BB.sc.BA,data_beans_BB.sc.SA, data_beans_BB.sc.TU)

#dim(data_beans_BB)
#dim(data_beans_BB.sc)

data_beans_BB<- cbind(data_beans_BB, data_beans_BB.sc)
#str(data_beans_BB)

#dim(data_beans_BB)

mod_MTME.bb <- asreml(fixed = cbind(GY, DM, PH, LD) ~ trait:rep + trait, # trait:rep -1,
                      random = ~ us(trait):name + us(trait):name:loc, #name:us(loc)
                      #residuals = ~dsum(~us(name)/loc),
                      residuals = ~ units:us(trait),
                      data = data_beans_BB,
                      trace = F,
                      maxit = 10000,
                      predict = predict.asreml(classify = "name:loc:trait") )

#update.asreml(mod_MTME.bb)
d1<- data.frame(wald(mod_MTME.bb))

d2<- data.frame(summary.asreml(mod_MTME.bb)$varcomp)

# results as data.table
blup.mtme.bb<- data.table((mod_MTME.bb$predictions$pvals[1:6])) # set the BLUP

# d1
#
#           Df Sum.of.Sq Wald.statistic Pr.Chisq.
# trait           4      0.323           0.323 9.88e-01
# trait:rep       12     55.248          55.248 1.63e-07
# residual (MS) NA      1.000             NA      NA

# d2
#
#           component std.error z.ratio bound X.ch
# trait:name!trait_GY:GY      0.25787      0.0798 3.2327 P 0.0
# trait:name!trait_DM:GY      0.11359      0.0692 1.6409 P 0.0
# trait:name!trait_DM:DM      0.36827      0.1091 3.3752 P 0.0
# trait:name!trait_PH:GY      0.08661      0.0496 1.7466 P 0.0
# trait:name!trait_PH:DM      0.09752      0.0570 1.7104 P 0.0
# trait:name!trait_PH:PH      0.11734      0.0545 2.1547 P 0.0
# trait:name!trait_LD:GY     -0.07238      0.0585 -1.2380 P 0.0
# trait:name!trait_LD:DM      0.05215      0.0652 0.7996 P 0.0
# trait:name!trait_LD:PH      0.01736      0.0451 0.3845 P 0.0
# trait:name!trait_LD:LD      0.15495      0.0707 2.1911 P 0.0
# trait:name:loc!trait_GY:GY      0.07426           NA      NA B 0.0
# trait:name:loc!trait_DM:GY    -0.00742           NA      NA B 0.0
# trait:name:loc!trait_DM:DM      0.07241           NA      NA B 0.0
# trait:name:loc!trait_PH:GY    -0.01275           NA      NA B 0.0

```

```

# trait:name:loc!trait_PH:DM  0.00149      NA      NA      B  0.0
# trait:name:loc!trait_PH:PH  0.06344      NA      NA      B  0.0
# trait:name:loc!trait_LD:GY  0.04234      NA      NA      B  0.0
# trait:name:loc!trait_LD:DM -0.00585      NA      NA      B  0.0
# trait:name:loc!trait_LD:PH -0.01030      NA      NA      B  0.0
# trait:name:loc!trait_LD:LD  0.09790      NA      NA      B  0.0
# units:trait!R              1.00000      NA      NA      F  0.0
# units:trait!trait_GY:GY     0.66440     0.0488 13.6191  P  0.0
# units:trait!trait_DM:GY     0.09813     0.0354  2.7707  P  0.0
# units:trait!trait_DM:DM     0.53533     0.0443 12.0755  P  0.0
# units:trait!trait_PH:GY     0.14825     0.0430  3.4472  P  0.0
# units:trait!trait_PH:DM     0.02495     0.0380  0.6564  P  0.0
# units:trait!trait_PH:PH     0.80001     0.0651 12.2932  P  0.0
# units:trait!trait_LD:GY     0.04123     0.0465  0.8857  P  0.0
# units:trait!trait_LD:DM     0.11857     0.0416  2.8483  P  0.0
# units:trait!trait_LD:PH     0.00104     0.0485  0.0214  P  0.2
# units:trait!trait_LD:LD     0.78932     0.0686 11.5024  P  0.0

```

5.7.2 Navy beans

```

data_beans_NB <- droplevels(data_beans_compl[which(data_beans_compl$mkt=="NB")])
#data_beans_NB <- na.omit(data_beans_NB)
#str(data_beans_NB)
#TU
data_beans_NB.sc.TU <- droplevels(subset(data_beans_NB, loc=="TU"))
data_beans_NB.sc.TU <-scale(data_beans_NB.sc.TU[,5:8])
#dim(data_beans_NB.sc.TU)
#BA
data_beans_NB.sc.BA <- droplevels(subset(data_beans_NB, loc=="BA"))
data_beans_NB.sc.BA <-scale(data_beans_NB.sc.BA[,5:8])

#SA
data_beans_NB.sc.SA <- droplevels(subset(data_beans_NB, loc=="SA"))
data_beans_NB.sc.SA <-scale(data_beans_NB.sc.SA[,5:8])

#data_beans_NB.sc<- scale(data_beans_NB[,5:6])
data_beans_NB<- data_beans_NB %>%
  dplyr::select(name, loc, mkt,rep)

data_beans_NB.sc<- rbind(data_beans_NB.sc.BA,data_beans_NB.sc.SA, data_beans_NB.sc.TU)

#dim(data_beans_NB)
#dim(data_beans_NB.sc)

data_beans_NB<- cbind(data_beans_NB, data_beans_NB.sc)
#str(data_beans_NB)

#data_beans_NB <- na.omit(data_beans_NB)

mod_MTME.nb <- asreml(fixed = cbind(GY, DM, PH, LD) ~ trait:rep + trait, # trait:rep -1,
  random = ~ us(trait):name + us(trait):name:loc, #name:us(loc)

```

```

#residuals = ~dsum(~us(name)/loc),
residuals = ~ units:us(trait),
data = data_beans_NB,
trace = F,
maxit = 10000,
predict = predict.asreml(classify = "name:loc:trait") )
#update.asreml(mod_MTME.nb)
#update.asreml(mod_MTME)
d1<- data.frame(wald(mod_MTME.nb))

d2<-data.frame(summary.asreml(mod_MTME.nb)$varcomp)

# results as data.table
blup.mtme.nb<- data.table((mod_MTME.nb$predictions$pvals[1:6])) # set the BLUP

# d1
#           Df Sum.of.Sq Wald.statistic Pr.Chisq.
# trait      4    0.663          0.663 9.56e-01
# trait:rep  12   63.914          63.914 4.33e-09
# residual (MS) NA    1.000             NA      NA
#
# d2
#           component std.error z.ratio bound X.ch
# trait:name!trait_GY:GY    0.28345    0.0865    3.278    P    0
# trait:name!trait_DM:GY    0.18415    0.0733    2.514    P    0
# trait:name!trait_DM:DM    0.33087    0.1036    3.195    P    0
# trait:name!trait_PH:GY    0.12097    0.0620    1.951    P    0
# trait:name!trait_PH:DM    0.06994    0.0646    1.083    P    0
# trait:name!trait_PH:PH    0.22001    0.0765    2.875    P    0
# trait:name!trait_LD:GY    0.15304    0.0820    1.867    P    0
# trait:name!trait_LD:DM    0.23406    0.0922    2.539    P    0
# trait:name!trait_LD:PH   -0.06152    0.0715   -0.860    P    0
# trait:name!trait_LD:LD    0.42545    0.1305    3.261    P    0
# trait:name:loc!trait_GY:GY  0.08495    0.0408    2.081    P    0
# trait:name:loc!trait_DM:GY -0.00607    0.0315   -0.192    P    0
# trait:name:loc!trait_DM:DM  0.09322    0.0442    2.107    P    0
# trait:name:loc!trait_PH:GY -0.01022    0.0325   -0.315    P    0
# trait:name:loc!trait_PH:DM  0.04445    0.0323    1.376    P    0
# trait:name:loc!trait_PH:PH  0.05325    0.0464    1.148    P    0
# trait:name:loc!trait_LD:GY -0.01023    0.0285   -0.359    P    0
# trait:name:loc!trait_LD:DM -0.03865    0.0282   -1.372    P    0
# trait:name:loc!trait_LD:PH -0.00769    0.0288   -0.267    P    0
# trait:name:loc!trait_LD:LD  0.03354    0.0351    0.956    P    0
# units:trait!R             1.00000             NA      NA    F    0
# units:trait!trait_GY:GY    0.61060    0.0475   12.845    P    0
# units:trait!trait_DM:GY   -0.03874    0.0355   -1.093    P    0
# units:trait!trait_DM:DM    0.54619    0.0461   11.851    P    0
# units:trait!trait_PH:GY    0.12173    0.0416    2.928    P    0
# units:trait!trait_PH:DM   -0.02890    0.0380   -0.760    P    0
# units:trait!trait_PH:PH    0.73727    0.0625   11.803    P    0
# units:trait!trait_LD:GY    0.01485    0.0377    0.393    P    0
# units:trait!trait_LD:DM    0.04894    0.0351    1.396    P    0

```

```
# units:trait!trait_LD:PH      -0.04851    0.0408  -1.190    P    0
# units:trait!trait_LD:LD      0.57031    0.0503  11.330    P    0
```

5.7.3 Red beans

```
## MTME mixed model Small Red - Pink Beans
# This model is not working! Too many missing genotypes and few genotypes available for
#  → thos market class.

data_beans_SR <- droplevels(data_beans_compl[which(data_beans_compl$mkt=="SR")])

#data_beans_SR <- na.omit(data_beans_SR)

#dim(data_beans_SR)
#TU
data_beans_SR.sc.TU <- droplevels(subset(data_beans_SR, loc=="TU"))
data_beans_SR.sc.TU <-scale(data_beans_SR.sc.TU[,5:8])
#dim(data_beans_SR.sc.TU)
#BA
data_beans_SR.sc.BA <- droplevels(subset(data_beans_SR, loc=="BA"))
data_beans_SR.sc.BA <-scale(data_beans_SR.sc.BA[,5:8])

#SA
data_beans_SR.sc.SA <- droplevels(subset(data_beans_SR, loc=="SA"))
data_beans_SR.sc.SA <-scale(data_beans_SR.sc.SA[,5:8])

#data_beans_SR.sc<- scale(data_beans_SR[,5:6])
data_beans_SR<- data_beans_SR %>%
  dplyr::select(name, loc, mkt,rep)
#dim(data_beans_SR)
data_beans_SR.sc<- rbind(data_beans_SR.sc.BA,data_beans_SR.sc.SA, data_beans_SR.sc.TU)

#dim(data_beans_SR)
#dim(data_beans_SR.sc)

data_beans_SR<- cbind(data_beans_SR, data_beans_SR.sc)
#str(data_beans_SR)

mod_MTME.sr <- asreml(fixed = cbind(GY, DM, PH, LD) ~ trait:rep + trait, # trait:rep -1,
  random = ~ us(trait):name + us(trait):name:loc, #name:us(loc)
  #residuals = ~dsum(~us(name)|loc),
  residuals = ~ units:us(trait),
  data = data_beans_SR,
  trace = F,
  maxit = 10000,
  predict = predict.asreml(classify = "name:loc:trait") )

#update.asreml(mod_MTME.sr)
d1<- data.frame(wald(mod_MTME.sr))

#
```



```
d2<- data.frame(summary.asreml(mod_MTME.sr)$varcomp)

# d1
#           Df Sum.of.Sq Wald.statistic Pr.Chisq.
# trait      4    0.0747         0.0747 9.99e-01
# trait:rep  12   58.7787         58.7787 3.77e-08
# residual (MS) NA     1.0000             NA     NA

# d2
#           component std.error z.ratio bound X.ch
# trait:name!trait_GY:GY    0.014673      NA      NA    B 0.0
# trait:name!trait_DM:GY    0.002083      NA      NA    B 0.0
# trait:name!trait_DM:DM    0.000615      NA      NA    B 0.0
# trait:name!trait_PH:GY    0.007643      NA      NA    B 0.0
# trait:name!trait_PH:DM    0.007450      NA      NA    B 0.0
# trait:name!trait_PH:PH    0.003715      NA      NA    B 0.0
# trait:name!trait_LD:GY   -0.020386      NA      NA    B 0.0
# trait:name!trait_LD:DM    0.015287      NA      NA    B 0.0
# trait:name!trait_LD:PH   -0.009582      NA      NA    B 0.0
# trait:name!trait_LD:LD    0.032827      NA      NA    B 0.0
# trait:name:loc!trait_GY:GY 0.257368      NA      NA    B 0.0
# trait:name:loc!trait_DM:GY -0.140886      NA      NA    B 0.0
# trait:name:loc!trait_DM:DM 0.427960      NA      NA    B 0.0
# trait:name:loc!trait_PH:GY -0.094343      NA      NA    B 0.0
# trait:name:loc!trait_PH:DM -0.293548      NA      NA    B 0.0
# trait:name:loc!trait_PH:PH 0.490821      NA      NA    B 0.0
# trait:name:loc!trait_LD:GY 0.257104      NA      NA    B 0.0
# trait:name:loc!trait_LD:DM -0.198700      NA      NA    B 0.0
# trait:name:loc!trait_LD:PH -0.051090      NA      NA    B 0.0
# trait:name:loc!trait_LD:LD 0.320474      NA      NA    B 0.0
# units:trait!R             1.000000      NA      NA    F 0.0
# units:trait!trait_GY:GY    0.754348    0.0923    8.173    P 0.0
# units:trait!trait_DM:GY    0.162496    0.0719    2.259    P 0.0
# units:trait!trait_DM:DM    0.792539    0.1027    7.714    P 0.0
# units:trait!trait_PH:GY    0.311365    0.0741    4.204    P 0.0
# units:trait!trait_PH:DM    0.046331    0.0705    0.657    P 0.0
# units:trait!trait_PH:PH    0.757329    0.1005    7.538    P 0.0
# units:trait!trait_LD:GY   -0.025897    0.0648   -0.400    P 0.1
# units:trait!trait_LD:DM    0.108011    0.0672    1.608    P 0.0
# units:trait!trait_LD:PH   -0.017539    0.0661   -0.265    P 0.0
# units:trait!trait_LD:LD    0.634711    0.0852    7.451    P 0.0
```

5.8 MTME Var Comp and corr

- Variance Component and correlations analysis

```
# Black
#traits
a=summary(mod_MTME.bb)$varcomp[1:10,1]

z=matrix(0, 4,4)
```

```

z[upper.tri(z)| row(z)==col(z)] <- a
cora=z/sqrt(diag(z)%*%t(diag(z)))
#cora

#int traits - locations
b=summary(mod_MTME.bb)$varcomp[11:20,1]

z=matrix(0, 4,4)
z[upper.tri(z)| row(z)==col(z)] <- b
corb=z/sqrt(diag(z)%*%t(diag(z)))
#corb

# Navys
#traits
c=summary(mod_MTME.nb)$varcomp[1:10,1]

z=matrix(0, 4,4)
z[upper.tri(z)| row(z)==col(z)] <- c
corc=z/sqrt(diag(z)%*%t(diag(z)))
#corc

#int traits - locations
d=summary(mod_MTME.nb)$varcomp[11:20,1]

z=matrix(0, 4,4)
z[upper.tri(z)| row(z)==col(z)] <- d
cord=z/sqrt(diag(z)%*%t(diag(z)))
#cord

corr=as.matrix(bdiag(cora,corb, corc, cord))

rownames(corr)=c(
  "BB_GY",
  "BB_DM",
  "BB_PH",
  "BB_LD",
  "BB_Loc-GY",
  "BB_Loc-DM",
  "BB_Loc-PH",
  "BB_Loc-LD",
  "NB_GY",
  "NB_DM",
  "NB_PH",
  "NB_LD",
  "NB_Loc-GY",
  "NB_Loc-DM",
  "NB_Loc-PH",
  "NB_Loc-LD")

colnames(corr)=rownames(corr)

corr1<- as_tibble(corr)

```

```

corr1<- corr1 %>%
  tidyr::pivot_longer(
    cols = everything(),
    names_to = "var",
    values_to = "corr")

corr1<-corr1 %>%
  dplyr::filter(corr != 0) %>%
  dplyr::filter(corr != 1)

names=c(
  "G GY-DM_BB",
  "G GY-PH_BB",
  "G GY-LD_BB",
  "G PH-DM_BB",
  "G DM-LD_BB",
  "G PH-LD_BB",
  "GxE GY-DM_BB",
  "GxE GY-PH_BB",
  "GxE GY-LD_BB",
  "GxE PH-DM_BB",
  "GxE DM-LD_BB",
  "GxE PH-LD_BB",

  "G GY-DM_NB",
  "G GY-PH_NB",
  "G GY-LD_NB",
  "G PH-DM_NB",
  "G DM-LD_NB",
  "G PH-LD_NB",
  "GxE GY-DM_NB",
  "GxE GY-PH_NB",
  "GxE GY-LD_NB",
  "GxE PH-DM_NB",
  "GxE DM-LD_NB",
  "GxE PH-LD_NB"

)

names2=c(
  "GY-DM",
  "GY-PH",
  "GY-LD",
  "PH-DM",
  "DM-LD",
  "PH-LD",

  "GY-DM",
  "GY-PH",
  "GY-LD",
  "PH-DM",
  "DM-LD",
  "PH-LD",

```

```

"GY-DM",
"GY-PH",
"GY-LD",
"PH-DM",
"DM-LD",
"PH-LD",

"GY-DM",
"GY-PH",
"GY-LD",
"PH-DM",
"DM-LD",
"PH-LD"
)

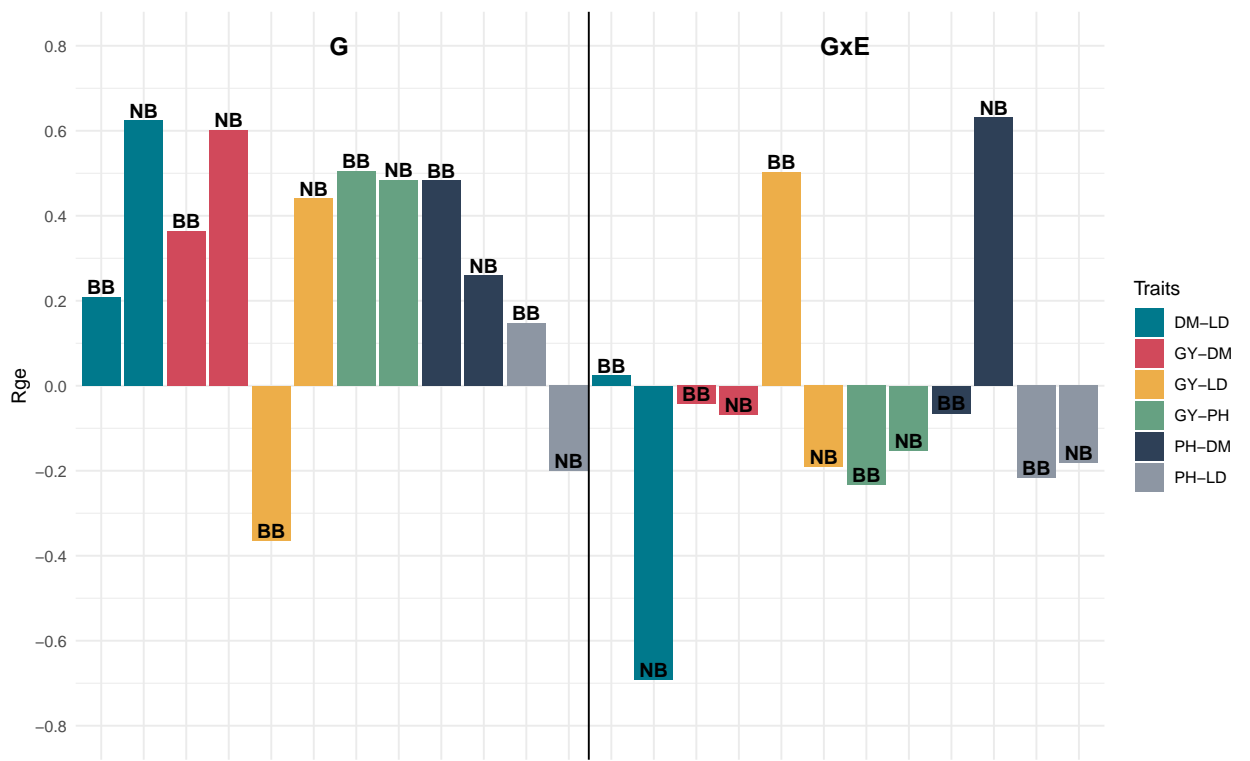
corr1$names<-names
corr1$names2<-names2

# Create new_id using dplyr only
corr1 <- corr1 %>%
  mutate(mkt = str_split(names, "_", simplify = TRUE)[ , 2])

#str(corr1)
branded_colors <- c("#00798c", "#d1495b", "#edae49",
                    "#66a182", "#2e4057", "#8d96a3")

ggplot(data=corr1, aes(x=names, y=corr, fill = names2)) +
  geom_bar(stat="identity", position=position_dodge())+
  geom_text(aes(label=mkt), vjust=-0.2, color="black",
            position = position_dodge(width = 1), size=4, fontface="bold")+
  scale_fill_brewer(palette="Paired")+
  theme_minimal() +
  theme(axis.text.x=element_blank(),
        strip.text=element_text(face="bold")) +
  scale_fill_manual(values=branded_colors) +
  labs(caption=NULL, x=NULL, y="Rge", fill = 'Traits') +
  scale_y_continuous(limits=c(-0.8, 0.8), n.breaks = 10) +
  geom_vline(xintercept = 12.47) +
  annotate(geom="text", x=6.6, y=0.8, label="G",size=5, fontface="bold",
          color="black") +
  annotate(geom="text", x=18.5, y=0.8, label="GxE",size=5, fontface="bold",
          color="black")

```



```
cat("\n\n\\pagebreak\n")
```

6 Package References

- Allaire J, Xie Y, McPherson J, Luraschi J, Ushey K, Atkins A, Wickham H, Cheng J, Chang W, Iannone R (2022). *rmarkdown: Dynamic Documents for R*. R package version 2.19, <https://github.com/rstudio/rmarkdown>. Xie Y, Allaire J, Golemund G (2018). *R Markdown: The Definitive Guide*. Chapman and Hall/CRC, Boca Raton, Florida. ISBN9781138359338, <https://bookdown.org/yihui/rmarkdown>. Xie Y, Dervieux C, Riederer E (2020). *R Markdown Cookbook*. Chapman and Hall/CRC, Boca Raton, Florida. ISBN 9780367563837, <https://bookdown.org/yihui/rmarkdown-cookbook>.
- Aphalo P (2022). *ggpmisc: Miscellaneous Extensions to 'ggplot2'*. R package version 0.5.1, <https://CRAN.R-project.org/package=ggpmisc>.
- Aphalo P (2022). *ggpp: Grammar Extensions to 'ggplot2'*. R package version 0.5.0, <https://CRAN.R-project.org/package=ggpp>.
- Auguie B (2017). *gridExtra: Miscellaneous Functions for "Grid" Graphics*. R package version 2.3, <https://CRAN.R-project.org/package=gridExtra>.
- Bache S, Wickham H (2022). *magrittr: A Forward-Pipe Operator for R*. R package version 2.0.3, <https://CRAN.R-project.org/package=magrittr>.
- Bates D, Maechler M, Jagan M (2022). *Matrix: Sparse and Dense Matrix Classes and Methods*. R package version 1.5-1, <https://CRAN.R-project.org/package=Matrix>.
- Becker OScbRA, Brownrigg. ARWRvbR (2022). *mapdata: Extra Map Databases*. R package version 2.3.1, <https://CRAN.R-project.org/package=mapdata>.
- Becker OScbRA, Minka ARWRvbRBEBTP, Deckmyn. A (2022). *maps: Draw Geographical Maps*. R package version 3.4.1, <https://CRAN.R-project.org/package=maps>.
- Bivand R, Nowosad J, Lovelace R (2022). *spData: Datasets for Spatial Analysis*. R package version 2.2.1, <https://CRAN.R-project.org/package=spData>.
- Brien C (2022). *asremlPlus: Augments 'ASReml-R' in Fitting Mixed Models and Packages Generally in Exploring Prediction Differences*. R package version 4.3.40, <https://CRAN.R-project.org/package=asremlPlus>.
- Butler D (2021). *asreml: Fits the Linear Mixed Model*. R package version 4.1.0.160, <www.vsnr.co.uk>.
- Dancho M, Vaughan D (2022). *tidyquant: Tidy Quantitative Financial Analysis*. R package version 1.0.6, <https://CRAN.R-project.org/package=tidyquant>.
- Dowle M, Srinivasan A (2022). *data.table: Extension of data.frame*. R package version 1.14.4, <https://CRAN.R-project.org/package=data.table>.
- Gohel D, Skintzos P (2022). *flextable: Functions for Tabular Reporting*. R package version 0.8.3, <https://CRAN.R-project.org/package=flextable>.
- Golemund G, Wickham H (2011). "Dates and Times Made Easy with lubridate." *Journal of Statistical Software*, 40(3), 1-25. <https://www.jstatsoft.org/v40/i03/>.
- Henry L, Wickham H (2022). *purrr: Functional Programming Tools*. R package version 0.3.5, <https://CRAN.R-project.org/package=purrr>.
- Hijmans R (2022). *raster: Geographic Data Analysis and Modeling*. R package version 3.6-3, <https://CRAN.R-project.org/package=raster>.
- Kassambara A (2022). *ggcorrplot: Visualization of a Correlation Matrix using 'ggplot2'*. R package version 0.1.4, <https://CRAN.R-project.org/package=ggcorrplot>.
- Kay M (2023). *ggdist: Visualizations of Distributions and Uncertainty*. doi:10.5281/zenodo.3879620 <https://doi.org/10.5281/zenodo.3879620>, R package version 3.2.1, <https://mjskay.github.io/ggdist/>.
- Müller K, Wickham H (2022). *tibble: Simple Data Frames*. R package version 3.1.8, <https://CRAN.R-project.org/package=tibble>.
- Nowosad J (2018). *'CARTOColors' Palettes*. R package version 1.0, <https://nowosad.github.io/rcartocolor>.
- Olivoto T, L'ucio AD (2020). "metan: an R package for multi-environment trial analysis." *Methods in Ecology and Evolution*, 11(6), 783-789. doi:10.1111/2041-210X.13384 <https://doi.org/10.1111/2041-210X.13384>.
- Patil I (2021). "Visualizations with statistical details: The 'ggstatsplot' approach." *Journal of Open Source Software*, 6(61), 3167. doi:10.21105/joss.03167 <https://doi.org/10.21105/joss.03167>, <https://>

- doi.org/10.21105/joss.03167.
- Pebesma E (2018). “Simple Features for R: Standardized Support for Spatial Vector Data.” *The R Journal*, 10(1), 439-446. doi:10.32614/RJ-2018-009 <https://doi.org/10.32614/RJ-2018-009>, <https://doi.org/10.32614/RJ-2018-009>.
 - Pebesma EJ, Bivand RS (2005). “Classes and methods for spatial data in R.” *R News*, 5(2), 9-13. <https://CRAN.R-project.org/doc/Rnews/>. Bivand RS, Pebesma E, Gomez-Rubio V (2013). *Applied spatial data analysis with R, Second edition*. Springer, NY. <https://asdar-book.org/>.
 - Pedersen T (2022). *ggforce: Accelerating ‘ggplot2’*. R package version 0.4.1, <https://CRAN.R-project.org/package=ggforce>.
 - Peterson BG, Carl P (2020). *PerformanceAnalytics: Econometric Tools for Performance and Risk Analysis*. R package version 2.0.4, <https://CRAN.R-project.org/package=PerformanceAnalytics>.
 - R Core Team (2022). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. <https://www.R-project.org/>.
 - Robinson D, Hayes A, Couch S (2022). *broom: Convert Statistical Objects into Tidy Tibbles*. R package version 1.0.2, <https://CRAN.R-project.org/package=broom>.
 - Ryan JA, Ulrich JM (2022). *quantmod: Quantitative Financial Modelling Framework*. R package version 0.4.20, <https://CRAN.R-project.org/package=quantmod>.
 - Ryan JA, Ulrich JM (2022). *xts: eXtensible Time Series*. R package version 0.12.2, <https://CRAN.R-project.org/package=xts>.
 - Schauburger P, Walker A (2022). *openxlsx: Read, Write and Edit xlsx Files*. R package version 4.2.5.1, <https://CRAN.R-project.org/package=openxlsx>.
 - Spinu V (2022). *timechange: Efficient Manipulation of Date-Times*. R package version 0.1.1, <https://CRAN.R-project.org/package=timechange>.
 - Tierney N, Cook D, McBain M, Fay C (2021). *nanianr: Data Structures, Summaries, and Visualisations for Missing Data*. R package version 0.6.1, <https://CRAN.R-project.org/package=nanianr>.
 - Ulrich J (2021). *TTR: Technical Trading Rules*. R package version 0.24.3, <https://CRAN.R-project.org/package=TTR>.
 - van Rossum B (2022). *statgenGxE: Genotype by Environment (GxE) Analysis*. R package version 1.0.5, <https://CRAN.R-project.org/package=statgenGxE>.
 - Walker K (2022). *tigris: Load Census TIGER/Line Shapefiles*. R package version 2.0, <https://CRAN.R-project.org/package=tigris>.
 - Wickham H (2011). “The Split-Apply-Combine Strategy for Data Analysis.” *Journal of Statistical Software*, 40(1), 1-29. <https://www.jstatsoft.org/v40/i01/>.
 - Wickham H (2016). *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York. ISBN 978-3-319-24277-4, <https://ggplot2.tidyverse.org>.
 - Wickham H (2022). *forcats: Tools for Working with Categorical Variables (Factors)*. R package version 0.5.2, <https://CRAN.R-project.org/package=forcats>.
 - Wickham H (2022). *stringr: Simple, Consistent Wrappers for Common String Operations*. R package version 1.5.0, <https://CRAN.R-project.org/package=stringr>.
 - Wickham H, Averick M, Bryan J, Chang W, McGowan LD, François R, Gromlund G, Hayes A, Henry L, Hester J, Kuhn M, Pedersen TL, Miller E, Bache SM, Müller K, Ooms J, Robinson D, Seidel DP, Spinu V, Takahashi K, Vaughan D, Wilke C, Woo K, Yutani H (2019). “Welcome to the tidyverse.” *Journal of Open Source Software*, 4(43), 1686. doi:10.21105/joss.01686 <https://doi.org/10.21105/joss.01686>.
 - Wickham H, François R, Henry L, Müller K (2022). *dplyr: A Grammar of Data Manipulation*. R package version 1.0.10, <https://CRAN.R-project.org/package=dplyr>.
 - Wickham H, Girlich M (2022). *tidyr: Tidy Messy Data*. R package version 1.2.1, <https://CRAN.R-project.org/package=tidyr>.
 - Wickham H, Hester J, Bryan J (2022). *readr: Read Rectangular Text Data*. R package version 2.1.3, <https://CRAN.R-project.org/package=readr>.
 - Wilke C (2020). *cowplot: Streamlined Plot Theme and Plot Annotations for ‘ggplot2’*. R package version 1.1.1, <https://CRAN.R-project.org/package=cowplot>.
 - Wolak ME (2012). “nadiv: an R package to create relatedness matrices for estimating non-additive genetic variances in animal models.” *Methods in Ecology and Evolution*, 3(5), 792-796.

- Xie Y, Cheng J, Tan X (2022). *DT: A Wrapper of the JavaScript Library ‘DataTables’*. R package version 0.26, <https://CRAN.R-project.org/package=DT>.
- Zeileis A, Grothendieck G (2005). “zoo: S3 Infrastructure for Regular and Irregular Time Series.” *Journal of Statistical Software*, 14(6), 1-27. doi:10.18637/jss.v014.i06 <https://doi.org/10.18637/jss.v014.i06>.
- Zhu H (2021). *kableExtra: Construct Complex Table with ‘kable’ and Pipe Syntax*. R package version 1.3.4, <https://CRAN.R-project.org/package=kableExtra>.

References

- Aguate, F., Crossa, J., & Balzarini, M. (2019). Effect of Missing Values on Variance Component Estimates in Multienvironment Trials. *Crop Science*, 59(2), 508–517. <https://doi.org/10.2135/cropsci2018.03.0209>
- Ahakupaz, F., Abdi, H., Neyestani, E., Hesami, A., Mohammadi, B., Mahmoudi, K. N., ... Alipour, H. (2021). Genotype-by-environment interaction analysis for grain yield of barley genotypes under dryland conditions and the role of monthly rainfall. *Agricultural Water Management*, 245, 106665. <https://doi.org/10.1016/j.agwat.2020.106665>
- Bänziger, M., & Cooper, M. (2001). Breeding for low input conditions and consequences for participatory plant breeding examples from tropical maize and wheat. *Euphytica*, 122:503519, 18. <https://doi.org/10.1023/A:1017510928038>
- Bates, D., Mächler, M., Bolker, B., & Walker, S. (2015). Fitting Linear Mixed-Effects Models Using **Lme4**. *Journal of Statistical Software*, 67(1). <https://doi.org/10.18637/jss.v067.i01>
- Bernardeli, A., Rocha, J. R. A. S. de C., Borém, A., Lorenzoni, R., Aguiar, R., Silva, J. N. B., ... Lamas Costa, M. D. (2021). Modeling spatial trends and enhancing genetic selection: An approach to soybean seed composition breeding. *Crop Science*, 61(2), 976–988. <https://doi.org/10.1002/csc2.20364>
- Bernardo, R. (2014). *Essentials of plant breeding*. Woodbury, Minnesota: Stemma Press.
- Boer, M. P., Wright, D., Feng, L., Podlich, D. W., Luo, L., Cooper, M., & van Eeuwijk, F. A. (2007). A Mixed-Model Quantitative Trait Loci (QTL) Analysis for Multiple-Environment Trial Data Using Environmental Covariables for QTL-by-Environment Interactions, With an Example in Maize. *Genetics*, 177(3), 1801–1813. <https://doi.org/10.1534/genetics.107.071068>
- Buntaran, H., Forkman, J., & Piepho, H.-P. (2021). Projecting results of zoned multi-environment trials to new locations using environmental covariates with random coefficient models: Accuracy and precision. *Theoretical and Applied Genetics*, 134(5), 1513–1530. <https://doi.org/10.1007/s00122-021-03786-2>
- Buntaran, H., Piepho, H., Schmidt, P., Rydén, J., Halling, M., & Forkman, J. (2020). Cross-validation of stagewise mixed-model analysis of Swedish variety trials with winter wheat and spring barley. *Crop Science*, 60(5), 2221–2240. <https://doi.org/10.1002/csc2.20177>
- Bustos-Korts, D., Boer, M. P., Layton, J., Gehringer, A., Tang, T., Wehrens, R., ... Van Eeuwijk, F. A. (2022). Identification of environment types and adaptation zones with self-organizing maps; applications to sunflower multi-environment data in Europe. *Theoretical and Applied Genetics*, 135(6), 2059–2082. <https://doi.org/10.1007/s00122-022-04098-9>
- Butler, D. G., Cullis, B. R., Gilmour, A. R., Gogel, B. J., & Thompson, R. (2017). *ASReml-R Reference Manual Version 4*. Hemel Hempstead, HP1 1ES, UK: VSN International Ltd.
- Cichy, K. A., Wiesinger, J. A., Berry, M., Nchimbi-Msolla, S., Fourie, D., Porch, T. G., ... Miklas, P. N. (2019). The role of genotype and production environment in determining the cooking time of dry beans (*PHASEOLUS VULGARIS* L.). *Legume Science*, 1(1). <https://doi.org/10.1002/leg3.13>
- Colombari Filho, J. M., de Resende, M. D. V., de Moraes, O. P., de Castro, A. P., Guimarães, É. P., Pereira, J. A., ... Breseghello, F. (2013). Upland rice breeding in Brazil: A simultaneous genotypic evaluation of stability, adaptability and grain yield. *Euphytica*, 192(1), 117–129. <https://doi.org/10.1007/s10681-013-0922-2>
- Cooper, M., Messina, C. D., Podlich, D., Totir, L. R., Baumgarten, A., Hausmann, N. J., ... Graham, G. (2014). Predicting the future of plant breeding: Complementing empirical evaluation with genetic prediction. *Crop and Pasture Science*, 65(4), 311. <https://doi.org/10.1071/CP14007>
- Costa-Neto, G. M. F., Moraes Júnior, O. P., Heinemann, A. B., de Castro, A. P., & Duarte, J. B. (2020). A novel GIS-based tool to reveal spatial trends in reaction norm: Upland rice case study. *Euphytica*, 216(3), 37. <https://doi.org/10.1007/s10681-020-2573-4>

- Costa-Neto, G., Fritsche-Neto, R., & Crossa, J. (2021). Nonlinear kernels, dominance, and envirotyping data increase the accuracy of genome-based prediction in multi-environment trials. *Heredity*, 126(1), 92–106. <https://doi.org/10.1038/s41437-020-00353-1>
- Costa-Neto, G., Galli, G., Carvalho, H. F., Crossa, J., & Fritsche-Neto, R. (n.d.). *EnvRtype: A software to interplay enviromics and quantitative genomics in agriculture*. 64.
- Crespo-Herrera, L., Howard, R., Piepho, H.-P., Pérez-Rodríguez, P., Montesinos-Lopez, O., Burgueño, J., ... Crossa, J. (2021). Genome-enabled prediction for sparse testing in multi-environmental wheat trials. *The Plant Genome*, 14(3), e20151. <https://doi.org/10.1002/tpg2.20151>
- Crossa, Jose, Fritsche-Neto, R., Montesinos-Lopez, O. A., Costa-Neto, G., Dreisigacker, S., Montesinos-Lopez, A., & Bentley, A. R. (2021). The Modern Plant Breeding Triangle: Optimizing the Use of Genomics, Phenomics, and Enviromics Data. *Frontiers in Plant Science*, 12, 651480. <https://doi.org/10.3389/fpls.2021.651480>
- Crossa, José, Yang, R.-C., & Cornelius, P. L. (2004). Studying crossover genotype \times environment interaction using linear-bilinear models and mixed models. *Journal of Agricultural, Biological, and Environmental Statistics*, 9(3), 362–380. <https://doi.org/10.1198/108571104X4423>
- da Silva, Karla Jorge, Teodoro, P. E., da Silva, M. J., Teodoro, L. P. R., Cardoso, M. J., Godinho, V. de P. C., ... Menezes, C. B. (2021). Identification of mega-environments for grain sorghum in Brazil using GGE biplot methodology. *Agronomy Journal*, 113(4), 3019–3030. <https://doi.org/10.1002/agj2.20707>
- Davis, W. V., & Lucier, Gary. (2021). *Vegetables and Pulses Outlook: November 2021* (No. VGS-367; p. 52). USDA.
- De La Fuente, G. N., Frei, U. K., & Lübberstedt, T. (2013). Accelerating plant breeding. *Trends in Plant Science*, 18(12), 667–672. <https://doi.org/10.1016/j.tplants.2013.09.001>
- DeLacy, I. H., Basford, K. E., Cooper, M., Bull, J. K., & McLaren, C. G. (1996). Analysis of multi-environment trials - an historical perspective. *Plant Adaptation and Crop Improvement.*, 39–124. Retrieved from <https://www.cabdirect.org/cabdirect/abstract/19961610824>
- Dramadri, I. O., Nkalubo, S. T., Kramer, D. M., & Kelly, J. D. (2021). Genome-wide association analysis of drought adaptive traits in common bean. *Crop Science*, 61(5), 3232–3253. <https://doi.org/10.1002/csc2.20484>
- Fehr, W. R. (n.d.). *Stages of Soybean Development*. 12.
- Ferreira, F. M., de Carvalho Rocha, J. R. do A. S., Alves, R. S., Malikouski, R. G., Peixoto, M. A., Suellen Sales de Oliveira, ... Bhering, L. L. (2021). GGE biplot-based genetic selection to guide interspecific crossing in *Corymbia* spp. *Bragantia*. <https://doi.org/10.1590/1678-4499.20210195>
- Finlay, K., & Wilkinson, G. (1963). The analysis of adaptation in a plant-breeding programme. *Australian Journal of Agricultural Research*, 14(6), 742. <https://doi.org/10.1071/AR9630742>
- Gabriel, K. R. (2002). Goodness of fit of biplots and correspondence analysis. *Biometrika*, 89(2), 423–436. <https://doi.org/10.1093/biomet/89.2.423>
- Galli, G., Horne, D. W., Collins, S. D., Jung, J., Chang, A., Fritsche-Neto, R., & Rooney, W. L. (2020). Optimization of UAS-based high-throughput phenotyping to estimate plant health and grain yield in sorghum. *The Plant Phenome Journal*, 3(1). <https://doi.org/10.1002/ppj2.20010>
- Gauch, H. G. (1988). Model Selection and Validation for Yield Trials with Interaction. *Biometrics*, 44(3), 705. <https://doi.org/10.2307/2531585>
- Gauch, H. G. (Ed.). (1992). *Statistical Analysis of Regional Yield Trials: AMMI Analysis of Factorial Designs* (1st edition). Amsterdam ; New York: Elsevier Science.
- Gauch, Hugh. G., & Zobel, R. W. (1997). Identifying Mega-Environments and Targeting Genotypes. *Crop Science*, 37(2), 311–326. <https://doi.org/10.2135/cropsci1997.0011183X003700020002x>
- Gilmour, A. R., Gogel, B. J., Cullis, B. R., Welham, S. J., & Thompson, R. (2015). *ASReml User Guide Release 4.1 Functional Specification* (HP1 1ES). Retrieved from <http://www.vsnr.co.uk/>
- González-Barrios, P., Díaz-García, L., & Gutiérrez, L. (2019). Mega-Environmental Design: Using Genotype \times Environment Interaction to Optimize Resources for Cultivar Testing. *Crop Science*, 59(5), 1899–1915. <https://doi.org/10.2135/cropsci2018.11.0692>
- Gurumurthy, S., Singh, J., Basu, P. S., Meena, S. K., Rane, J., Singh, N. P., & Hazra, K. K. (2022). Increased significance of chickpea (*Cicer arietinum* L.) Senescence trait under water-deficit environment. *Environmental Challenges*, 8, 100565. <https://doi.org/10.1016/j.envc.2022.100565>
- Hamblin, J., & Zimmermann, M. J. de O. (1986). Breeding Common Bean for Yield in Mixtures. In J.

- Janick (Ed.), *Plant Breeding Reviews* (pp. 245–272). <https://doi.org/10.1002/9781118061015.ch8>
- Happ, M. M., Graef, G. L., Wang, H., Howard, R., Posadas, L., & Hyten, D. L. (2021). Comparing a Mixed Model Approach to Traditional Stability Estimators for Mapping Genotype by Environment Interactions and Yield Stability in Soybean [*Glycine max* (L.) Merr.]. *Frontiers in Plant Science*, 12, 630175. <https://doi.org/10.3389/fpls.2021.630175>
- Hartung, J., & Piepho, H. (2021). Effect of missing values in multi-environmental trials on variance component estimates. *Crop Science*, 61(6), 4087–4097. <https://doi.org/10.1002/csc2.20621>
- Heilig, J. A., Wright, E. M., & Kelly, J. D. (2017). Symbiotic Nitrogen Fixation of Black and Navy Bean under Organic Production Systems. *Agronomy Journal*, 109(5), 2223–2230. <https://doi.org/10.2134/agronj2017.01.0051>
- Henderson, C. R. (1975). Best Linear Unbiased Estimation and Prediction under a Selection Model. *Biometrics*, 31(2), 423. <https://doi.org/10.2307/2529430>
- Hoyos-Villegas, V., O'Connor, J. R., Heslop, A. D., Hilditch, A., Jahufer, M. Z. Z., & Barrett, B. A. (2019). Rate of Genetic Gain for Persistence to Grazing and Dry Matter Yield in White Clover across 90 Years of Cultivar Development. *Crop Science*, 59(2), 537–552. <https://doi.org/10.2135/cropsci2018.07.0471>
- Hoyos-Villegas, V., Wright, E. M., & Kelly, J. D. (2016). GGE Biplot Analysis of Yield Associations with Root Traits in a Mesoamerican Bean Diversity Panel. *Crop Science*, 56(3), 1081–1094. <https://doi.org/10.2135/cropsci2015.10.0609>
- Huang, X., Jang, S., Kim, B., Piao, Z., Redona, E., & Koh, H.-J. (2021). Evaluating Genotype \times Environment Interactions of Yield Traits and Adaptability in Rice Cultivars Grown under Temperate, Subtropical and Tropical Environments. *Agriculture*, 11(6), 558. <https://doi.org/10.3390/agriculture11060558>
- Isik, F., Holland, J., & Maltecca, C. (2017). *Genetic Data Analysis for Plant and Animal Breeding*. <https://doi.org/10.1007/978-3-319-55177-7>
- Kamfwa, K., Cichy, K. A., & Kelly, J. D. (2015). Genome-Wide Association Study of Agronomic Traits in Common Bean. *The Plant Genome*, 8(2). <https://doi.org/10.3835/plantgenome2014.09.0059>
- Kang, M. S. (2020). Genotype-environment interaction and stability analyses: An update. In M. S. Kang (Ed.), *Quantitative genetics, genomics and plant breeding* (2nd ed., pp. 140–161). <https://doi.org/10.1079/9781789240214.0140>
- Katuramu, D. N., Luyima, G. B., Nkalubo, S. T., Wiesinger, J. A., Kelly, J. D., & Cichy, K. A. (2020). On-farm multi-location evaluation of genotype by environment interactions for seed yield and cooking time in common bean. *Scientific Reports*, 10(1), 3628. <https://doi.org/10.1038/s41598-020-60087-2>
- Katuramu, D. N., Wiesinger, J. A., Luyima, G. B., Nkalubo, S. T., Glahn, R. P., & Cichy, K. A. (2021). Investigation of Genotype by Environment Interactions for Seed Zinc and Iron Concentration and Iron Bioavailability in Common Bean. *Frontiers in Plant Science*, 12, 670965. <https://doi.org/10.3389/fpls.2021.670965>
- Kelly, James D., Awale, H. E., Wiersma, A. T., & Wright, E. M. (2021). Registration of “Adams” black bean. *Journal of Plant Registrations*, 15(2), 253–259. <https://doi.org/10.1002/plr2.20063>
- Kelly, James D., & Cichy, K. A. (2012). Dry Bean Breeding and Production Technologies. In M. Siddiq & M. A. Uebersax (Eds.), *Dry Beans and Pulses Production, Processing and Nutrition* (1st ed., pp. 23–54). <https://doi.org/10.1002/9781118448298.ch2>
- Kelly, James D., Kolkman, J. M., & Schneider, K. (1998). Breeding for yield in dry bean (*Phaseolus vulgaris* L.). *Euphytica*, 102(343–356), 14. <https://doi.org/10.1023/A:1018392901978>
- Kelly, James D., Varner, G. V., Cichy, K. A., & Wright, E. M. (2015). Registration of “Alpena” Navy Bean. *Journal of Plant Registrations*, 9(1), 10–14. <https://doi.org/10.3198/jpr2014.04.0025crc>
- Koundinya A.V.V., Ajeesh B.R., Vivek H., Sheela M.N., Mohan C., & Asha K.I. (2021). Genetic parameters, stability and selection of cassava genotypes between rainy and water stress conditions using AMMI, WAAS, BLUP and MTSI. *Scientia Horticulturae*, 281, 109949. <https://doi.org/10.1016/j.scienta.2021.109949>
- Krause, M. D., Dias, K. O. G., Singh, A. K., & Beavis, W. D. (2022). *Using large soybean historical data to study genotype by environment variation and identify mega-environments with the integration of genetic and non-genetic factors* [Preprint]. <https://doi.org/10.1101/2022.04.11.487885>
- Ligarreto-Moreno, G., & Pimentel-Ladino, C. (2022). Grain yield and genotype \times environment interaction in bean cultivars with different growth habits. *Plant Production Science*, 25(2), 232–241. <https://doi.org/10.1080/1343943X.2021.1981141>

- Lin, C. S., & Binns, M. R. (1988). A SUPERIORITY MEASURE OF CULTIVAR PERFORMANCE FOR CULTIVAR \times LOCATION DATA. *Canadian Journal of Plant Science*, 68(1), 193–198. <https://doi.org/10.4141/cjps88-018>
- MacQueen, A. H., White, J. W., Lee, R., Osorno, J. M., Schmutz, J., Miklas, P. N., ... Juenger, T. E. (2020). Genetic Associations in Four Decades of Multienvironment Trials Reveal Agronomic Trait Evolution in Common Bean. *Genetics*, 215(1), 267–284. <https://doi.org/10.1534/genetics.120.303038>
- Malosetti, M., Bustos-Korts, D., Boer, M. P., & van Eeuwijk, F. A. (2016). Predicting Responses in Multiple Environments: Issues in Relation to Genotype \times Environment Interactions. *Crop Science*, 56(5), 2210–2222. <https://doi.org/10.2135/cropsci2015.05.0311>
- Malosetti, M., Ribaut, J.-M., & van Eeuwijk, F. A. (2013). The statistical analysis of multi-environment data: Modeling genotype-by-environment interaction and its genetic basis. *Frontiers in Physiology*, 4. <https://doi.org/10.3389/fphys.2013.00044>
- Miklas, P. N., Kelly, J. D., Beebe, S. E., & Blair, M. W. (2006). Common bean breeding for resistance against biotic and abiotic stresses: From classical to MAS breeding. *Euphytica*, 147(1), 105–131. <https://doi.org/10.1007/s10681-006-4600-5>
- Miklas, P. N., Kelly, J. D., & Cichy, K. A. (2022). Dry Bean Breeding and Production Technologies. In M. Siddiq & M. A. Uebersax (Eds.), *Dry Beans and Pulses* (1st ed., pp. 29–56). <https://doi.org/10.1002/9781119776802.ch2>
- Millet, E., Welcker, C., Kruijer, W., Negro, S., Nicolas, S., Praud, S., ... Bauland, C. (2016). Genome-wide analysis of yield in Europe: Allelic effects as functions of drought and heat scenarios. *Plant Physiology*, pp.00621.2016. <https://doi.org/10.1104/pp.16.00621>
- Mrode, R. A. (2014). *Linear models for the prediction of animal breeding values* (3rd ed). Boston, MA: CABI.
- Nataraj, V., Bhartiya, A., Singh, C. P., Devi, H. N., Deshmukh, M. P., Verghese, P., ... Gupta, S. (2021). WAASB-based stability analysis and simultaneous selection for grain yield and early maturity in soybean. *Agronomy Journal*, 113(4), 3089–3099. <https://doi.org/10.1002/agj2.20750>
- Olivoto, T., Lúcio, A. D. C., Silva, J. A. G., Marchioro, V. S., Souza, V. Q., & Jost, E. (2019). Mean Performance and Stability in Multi-Environment Trials I: Combining Features of AMMI and BLUP Techniques. *Agronomy Journal*, 111(6), 2949–2960. <https://doi.org/10.2134/agronj2019.03.0220>
- Olivoto, T., Lúcio, A. D. C., Silva, J. A. G., Sari, B. G., & Diel, M. I. (2019). Mean Performance and Stability in Multi-Environment Trials II: Selection Based on Multiple Traits. *Agronomy Journal*, 111(6), 2961–2969. <https://doi.org/10.2134/agronj2019.03.0221>
- Olivoto, T., Nardino, M., Meira, D., Meier, C., Follmann, D. N., Souza, V. Q., ... Baretta, D. (2021). Multi-trait selection for mean performance and stability in maize. *Agronomy Journal*, 113(5), 3968–3974. <https://doi.org/10.1002/agj2.20741>
- Philipo, M., Ndakidemi, P. A., & Mbega, E. R. (2021). Environmentally stable common bean genotypes for production in different agro-ecological zones of Tanzania. *Heliyon*, 7(1), e05973. <https://doi.org/10.1016/j.heliyon.2021.e05973>
- Piepho, H.-P. (2018). Allowing for the structure of a designed experiment when estimating and testing trait correlations. *The Journal of Agricultural Science*, 156(1), 59–70. <https://doi.org/10.1017/S0021859618000059>
- Piepho, H.-P., Edmondson, R., & Yaseen, M. (n.d.). *Tutorial Analysis of Some Agricultural Experiments*. 46.
- Pour-Aboughadareh, A., Khalili, M., Pocza, P., & Olivoto, T. (2022). Stability Indices to Deciphering the Genotype-by-Environment Interaction (GEI) Effect: An Applicable Review for Use in Plant Breeding Programs. *Plants*, 11(3), 414. <https://doi.org/10.3390/plants11030414>
- Qureshi, A. M. I., & Sadohara, R. (2019). Breeding Dry Beans (*Phaseolus vulgaris* L.) With Improved Cooking and Canning Quality Traits. In A. M. I. Qureshi, Z. A. Dar, & S. H. Wani (Eds.), *Quality Breeding in Field Crops* (pp. 173–191). https://doi.org/10.1007/978-3-030-04609-5_8
- Rakshit, S., K N, G., Gomashe, S., Swapna, M., More, A., Gadakh, S., ... Bhat, P. (2014). GGE biplot analysis of genotype \times environment interaction in rabi grain sorghum [*Sorghum bicolor* (L.) Moench]. *Indian Journal of Genetics and Plant Breeding*, 74, 558–563. <https://doi.org/10.5958/0975-6906.2014.00889.X>
- Resende, M. D. V. de. (2016). Software Selegen-REML/BLUP: A useful tool for plant breeding. *Crop*

- Breeding and Applied Biotechnology*, 16(4), 330–339. <https://doi.org/10.1590/1984-70332016v16n4a49>
- Resende, R. T., Piepho, H.-P., Rosa, G. J. M., Silva-Junior, O. B., e Silva, F. F., de Resende, M. D. V., & Grattapaglia, D. (2021). Enviromics in breeding: Applications and perspectives on envirotypic-assisted selection. *Theoretical and Applied Genetics*, 134(1), 95–112. <https://doi.org/10.1007/s00122-020-03684-z>
- Rodrigues, F. C., Silva, F. C. S., Carneiro, P. C. S., Peternelli, L. A., Bhering, L. L., & da Silva, F. L. (2022). Environmental stratification in trials of unbalanced multiyear soybean (*Glycine max* (L.) Merrill) via the integration of GGE Biplot graphs and networks of environmental similarity. *Euphytica*, 218(6), 71. <https://doi.org/10.1007/s10681-022-02994-1>
- Rosales, M. A., Ocampo, E., Rodríguez-Valentín, R., Olvera-Carrillo, Y., Acosta-Gallegos, J., & Covarrubias, A. A. (2012). Physiological analysis of common bean (*Phaseolus vulgaris* L.) Cultivars uncovers characteristics related to terminal drought resistance. *Plant Physiology and Biochemistry*, 56, 24–34. <https://doi.org/10.1016/j.plaphy.2012.04.007>
- Shukla, G. K. (1972). Some statistical aspects of partitioning genotype-environment components of variability. *Heredity*, 29(2), 237–245. <https://doi.org/10.1038/hdy.1972.87>
- Siddiq, M., & Uebersax, M. A. (Eds.). (2022). *Dry Beans and Pulses Production, Processing, and Nutrition* (Second Edition). <https://doi.org/10.1002/9781119776802.fmatter>
- Silva, Karla Jorge, Teodoro, P. E., Silva, M. J., Teodoro, L. P. R., Cardoso, M. J., Godinho, V. de P. C., ... Menezes, C. B. (2021). Identification of mega-environments for grain sorghum in Brazil using GGE biplot methodology. *Agronomy Journal*, 113(4), 3019–3030. <https://doi.org/10.1002/agj2.20707>
- Singamsetti, A., Shahi, J. P., Zaidi, P. H., Seetharam, K., Vinayan, M. T., Kumar, M., ... Madankar, K. (2021). Genotype \times environment interaction and selection of maize (*Zea mays* L.) Hybrids across moisture regimes. *Field Crops Research*, 270, 108224. <https://doi.org/10.1016/j.fcr.2021.108224>
- Singh, B., Das, A., Parihar, A. K., Bhagawati, B., Singh, D., Pathak, K. N., ... Gupta, S. (2020). Delineation of Genotype-by-Environment interactions for identification and validation of resistant genotypes in mungbean to root-knot nematode (*Meloidogyne incognita*) using GGE biplot. *Scientific Reports*, 10(1), 4108. <https://doi.org/10.1038/s41598-020-60820-x>
- Singh, S. P., Terán, H., Lema, M., Webster, D. M., Strausbaugh, C. A., Miklas, P. N., ... Brick, M. A. (2007). Seventy-five Years of Breeding Dry Bean of the Western USA. *Crop Science*, 47(3), 981–989. <https://doi.org/10.2135/cropsci2006.05.0322>
- Smith, A., Norman, A., Kuchel, H., & Cullis, B. (2021). Plant Variety Selection Using Interaction Classes Derived From Factor Analytic Linear Mixed Models: Models With Independent Variety Effects. *Frontiers in Plant Science*, 12, 737462. <https://doi.org/10.3389/fpls.2021.737462>
- Smith, M. R., Dinglasan, E., Veneklaas, E., Polania, J., Rao, I. M., Beebe, S. E., & Merchant, A. (2022). Effect of Drought and Low P on Yield and Nutritional Content in Common Bean. *Frontiers in Plant Science*, 13, 814325. <https://doi.org/10.3389/fpls.2022.814325>
- Tize, I., Fotso, A. K., Nukenine, E. N., Masso, C., Ngome, F. A., Suh, C., ... Hanna, R. (2021). New cassava germplasm for food and nutritional security in Central Africa. *Scientific Reports*, 11(1), 7394. <https://doi.org/10.1038/s41598-021-86958-w>
- Tokatlidis, I. S., Papadopoulos, I. I., Baxevanos, D., & Koutita, O. (2010). Genotype \times Environment Effects on Single-Plant Selection at Low Density for Yield and Stability in Climbing Dry Bean Populations. *Crop Science*, 50(3), 775–783. <https://doi.org/10.2135/cropsci2009.08.0459>
- Uebersax, M. A., Cichy, K. A., Gomez, F. E., Porch, T. G., Heitholt, J., Osorno, J. M., ... Bales, S. (2022). Dry beans (*Phaseolus vulgaris* L.) As a vital component of sustainable agriculture and food security—A review. *Legume Science*. <https://doi.org/10.1002/leg3.155>
- USDA, United States Department of Agriculture, & NASS, National Agricultural Statistics Service. (2022). *Crop Values Summary 2021*. 48. Retrieved from <https://usda.library.cornell.edu/concern/publications/k35694332>
- van Eeuwijk, F. A., Bustos-Korts, D. V., & Malosetti, M. (2016). What Should Students in Plant Breeding Know About the Statistical Aspects of Genotype \times Environment Interactions? *Crop Science*, 56(5), 2119–2140. <https://doi.org/10.2135/cropsci2015.06.0375>
- Vandemark, G. J., Brick, M. A., Osorno, J. M., Kelly, J. D., & Urrea, C. A. (2015). Edible Grain Legumes. In S. Smith, B. Diers, J. Specht, & B. Carver (Eds.), *CSSA Special Publications* (pp. 87–123). <https://doi.org/10.2135/cssaspecpub33.c5>
- Vandemark, G., Brick, M., Kelly, J., Osorno, J., & Urrea, C. (2017). YIELD GAINS IN DRY BEANS IN

- THE U.S. *Publications from USDA-ARS / UNL Faculty*. Retrieved from <https://digitalcommons.unl.edu/usdaarsfacpub/1781>
- Verma, A., & Singh, G. P. (2021). Stability, Adaptability Analysis of Wheat Genotypes by AMMI with BLUP for Restricted Irrigated Multi Location Trials in Peninsular Zone of India. *Agricultural Sciences*, 12(03), 198–212. <https://doi.org/10.4236/as.2021.123013>
- Volpato, L., Alves, R. S., Teodoro, P. E., Vilela de Resende, M. D., Nascimento, M., Nascimento, A. C. C., ... Borém, A. (2019). Multi-trait multi-environment models in the genetic selection of segregating soybean progeny. *PLOS ONE*, 14(4), e0215315. <https://doi.org/10.1371/journal.pone.0215315>
- Wang, W., Wright, E. M., Uebersax, M. A., & Cichy, K. (2021). A pilot-scale dry bean canning and evaluation protocol. *Journal of Food Processing and Preservation*, 00(e16171), e16171. <https://doi.org/10.1111/jfpp.16171>
- Yan, W. (2002). Singular-Value Partitioning in Biplot Analysis of Multi-environment Trial Data. *Agronomy Journal*, 94(5), 990. <https://doi.org/10.2134/agronj2002.0990>
- Yan, W. (2015). Mega-environment Analysis and Test Location Evaluation Based on Unbalanced Multiyear Data. *Crop Science*, 55(1), 113–122. <https://doi.org/10.2135/cropsci2014.03.0203>
- Yan, W. (2016). Analysis and Handling of $G \times E$ in a Practical Breeding Program. *Crop Science*, 56(5), 2106–2118. <https://doi.org/10.2135/cropsci2015.06.0336>
- Yan, W., Hunt, L. A., Sheng, Q., & Szlavnick, Z. (2000). Cultivar Evaluation and Mega-Environment Investigation Based on the GGE Biplot. *Crop Science*, 40(3), 597–605. <https://doi.org/10.2135/cropsci2000.403597x>
- Yan, W., & Kang, M. S. (2003). *GGE biplot analysis: A graphical tool for breeders, geneticists, and agronomists*. Boca Raton, Fla: CRC Press.
- Yan, W., Kang, M. S., Ma, B., Woods, S., & Cornelius, P. L. (2007). GGE Biplot vs. AMMI Analysis of Genotype-by-Environment Data. *Crop Science*, 47(2), 643–653. <https://doi.org/10.2135/cropsci2006.06.0374>
- Yan, W., & Tinker, N. A. (2006). Biplot analysis of multi-environment trial data: Principles and applications. *Canadian Journal of Plant Science*, 86(3), 623–645. <https://doi.org/10.4141/P05-169>
- Yohannes, S., Loha, G., & Gessese, M. K. (2020). Performance Evaluation of Common Bean (*Phaseolus Vulgaris* L.) Genotypes for Yield and Related Traits at Areka, Southern Ethiopia. *Advances in Agriculture*, 2020, 1–8. <https://doi.org/10.1155/2020/1497530>
- Zuffo, A. M., Steiner, F., Aguilera, J. G., Teodoro, P. E., Teodoro, L. P. R., & Busch, A. (2020). Multi-trait stability index: A tool for simultaneous selection of soya bean genotypes in drought and saline stress. *Journal of Agronomy and Crop Science*, 206(6), 815–822. <https://doi.org/10.1111/jac.12409>
- Zuiderveen, G. H., Padder, B. A., Kamfwa, K., Song, Q., & Kelly, J. D. (2016). Genome-Wide Association Study of Anthracnose Resistance in Andean Beans (*Phaseolus vulgaris*). *PLOS ONE*, 11(6), e0156391. <https://doi.org/10.1371/journal.pone.0156391>