

Here comes the mixed effects

Data

Download the data here: <https://shorturl.at/ehlX1>

Or read it directly into R using the above link

```
library(tidyverse)
strain <- read_csv("https://shorturl.at/ehlX1")
```

What are the variables in the data?

```
strain |>  
  glimpse()
```

Rows: 31,391

Columns: 14

```
$ game_id      <dbl> 2021090900, 2021090900, 2021090900, 2021090900, 20210~  
$ play_id     <dbl> 97, 97, 97, 137, 137, 137, 137, 187, 187, 187, 282, 2~  
$ rusher_id   <dbl> 42403, 44955, 53441, 35441, 35454, 40074, 46081, 3544~  
$ rusher_name <chr> "Randy Gregory", "Carlos Watkins", "Micah Parsons", "~  
$ rusher_position <chr> "DE", "DE", "OLB", "DE", "DE", "DE", "NT", "DE", "DE"~  
$ defensive_team <chr> "DAL", "DAL", "DAL", "TB", "TB", "TB", "TB", "TB", "T~  
$ blocker_id  <dbl> 42377, 42404, 46163, 46119, 37082, 42654, 47873, 4787~  
$ blocker_position <chr> "T", "G", "G", "G", "T", "T", "G", "G", "T", "C", "G"~  
$ offensive_team <chr> "TB", "TB", "TB", "DAL", "DAL", "DAL", "DAL", "DAL", "~  
$ n_blockers  <dbl> 1, 2, 2, 2, 1, 1, 1, 2, 1, 2, 2, 2, 1, 1, 1, 1, 2, 1,~  
$ down        <dbl> 3, 3, 3, 1, 1, 1, 1, 2, 2, 2, 1, 1, 1, 1, 1, 3, 3, 3,~  
$ yards_first_down <dbl> 2, 2, 2, 10, 10, 10, 10, 6, 6, 6, 10, 10, 10, 10, 10,~  
$ yards_endzone <dbl> 43, 43, 43, 108, 108, 108, 108, 76, 76, 76, 49, 49, 4~  
$ avg_strain  <dbl> 2.6411568, -1.8707514, 2.8472365, 2.2436906, 1.168554~
```

Model specification

Response: average STRAIN for pass rusher at the play-level

Random effects: pass rusher, pass blocker, defensive team, offensive team

Fixed effects:

- ▶ rusher position: DE*, OLB, DT, NT, interior, secondary
- ▶ blocker position: T*, C, G, other
- ▶ number of blockers involved in the play
- ▶ play-context (down, yards to go for a first down, yards from end zone)

[*] Reference level

Fitting a mixed effects model

```
# reorder factor levels - reference level
strain <- strain |>
  mutate(rusher_position = fct_relevel(rusher_position, "DE"),
         blocker_position = fct_relevel(blocker_position, "T"),
         down = fct_relevel(factor(down), "1"))

library(lme4)
strain_lmer <- lmer(avg_strain ~ (1 | rusher_id) + (1 | blocker_id) +
                   (1 | defensive_team) + (1 | offensive_team) +
                   rusher_position + blocker_position + n_blockers +
                   down + yards_first_down + yards_endzone,
                   data = strain)
# strain_lmer |> summary()
```

Inference

broom makes everything nicer and easier (Who originally created broom?
arxiv.org/abs/1412.3565)

broom.mixed: extension for mixed effects models

```
library(broom.mixed)
strain_lmer |>
  tidy(conf.int = TRUE, conf.level = 0.95)
# 3 broom verbs: tidy, glance, augment
# can also specify conf.method
# there are multiple methods
# help(lme4::confint.merMod)
```

Inference

	effect	group	term	estimate	std.error	statistic	conf.low	conf.high
1	fixed	<NA>	(Intercept)	1.8004	0.0991	18.1722	1.6062	1.9946
2	fixed	<NA>	rusher_positionDT	-0.7166	0.0678	-10.5658	-0.8495	-0.5836
3	fixed	<NA>	rusher_positioninteriorLB	0.5611	0.0999	5.6143	0.3653	0.7570
4	fixed	<NA>	rusher_positionNT	-0.9426	0.1021	-9.2344	-1.1427	-0.7425
5	fixed	<NA>	rusher_positionOLB	0.5574	0.0726	7.6742	0.4151	0.6998
6	fixed	<NA>	rusher_positionsecondary	1.4252	0.1114	12.7899	1.2068	1.6436
7	fixed	<NA>	blocker_positionC	-0.2983	0.0594	-5.0236	-0.4146	-0.1819
8	fixed	<NA>	blocker_positionG	-0.1546	0.0471	-3.2846	-0.2469	-0.0624
9	fixed	<NA>	blocker_positionother	-0.1260	0.0646	-1.9507	-0.2526	0.0006
10	fixed	<NA>	n_blockers	-0.7366	0.0259	-28.4153	-0.7875	-0.6858
11	fixed	<NA>	down0	-0.1384	0.1797	-0.7699	-0.4906	0.2139
12	fixed	<NA>	down2	0.5143	0.0290	17.7439	0.4575	0.5711
13	fixed	<NA>	down3	0.9176	0.0324	28.3613	0.8542	0.9810
14	fixed	<NA>	down4	0.4649	0.0767	6.0637	0.3146	0.6152
15	fixed	<NA>	yards_first_down	0.0491	0.0033	14.9013	0.0426	0.0555
16	fixed	<NA>	yards_endzone	0.0001	0.0005	0.1844	-0.0009	0.0011
17	ran_pars	rusher_id	sd__(Intercept)	0.4088	NA	NA	NA	NA
18	ran_pars	blocker_id	sd__(Intercept)	0.2121	NA	NA	NA	NA
19	ran_pars	defensive_team	sd__(Intercept)	0.2477	NA	NA	NA	NA
20	ran_pars	offensive_team	sd__(Intercept)	0.2555	NA	NA	NA	NA
21	ran_pars	Residual	sd__Observation	2.0594	NA	NA	NA	NA

If you really want p -values...

```
# Note: lmerTest masks every function in lme4 which is annoying  
library(lmerTest)  
# do this to avoid overriding existing models fitted with lme4  
strain_lmer_test <- strain_lmer |>  
  as("merModLmerTest")  
# strain_lmer_test |> summary()
```

```
strain_lmer_test |>  
  tidy()  
  
# by default  
# strain_lmer_test |> tidy(ddf.method = "Satterthwaite")  
# there are other methods  
# help(summary.lmerModLmerTest)
```


If you really want p -values...

this matches Table 6 in arxiv.org/pdf/2305.10262.pdf

	effect	group	term	estimate	std.error	statistic	df	p.value
1	fixed	<NA>	(Intercept)	1.8004	0.0991	18.1722	233.2409	0.0000
2	fixed	<NA>	rusher_positionDT	-0.7166	0.0678	-10.5658	319.1122	0.0000
3	fixed	<NA>	rusher_positioninteriorLB	0.5611	0.0999	5.6143	798.3171	0.0000
4	fixed	<NA>	rusher_positionNT	-0.9426	0.1021	-9.2344	317.2813	0.0000
5	fixed	<NA>	rusher_positionOLB	0.5574	0.0726	7.6742	360.6533	0.0000
6	fixed	<NA>	rusher_positionsecondary	1.4252	0.1114	12.7899	2019.3243	0.0000
7	fixed	<NA>	blocker_positionC	-0.2983	0.0594	-5.0236	185.3300	0.0000
8	fixed	<NA>	blocker_positionG	-0.1546	0.0471	-3.2846	169.5715	0.0012
9	fixed	<NA>	blocker_positionother	-0.1260	0.0646	-1.9507	955.2276	0.0514
10	fixed	<NA>	n_blockers	-0.7366	0.0259	-28.4153	30950.3242	0.0000
11	fixed	<NA>	down0	-0.1384	0.1797	-0.7699	30530.5280	0.4414
12	fixed	<NA>	down2	0.5143	0.0290	17.7439	30580.3644	0.0000
13	fixed	<NA>	down3	0.9176	0.0324	28.3613	30828.8115	0.0000
14	fixed	<NA>	down4	0.4649	0.0767	6.0637	30736.3364	0.0000
15	fixed	<NA>	yards_first_down	0.0491	0.0033	14.9013	30722.3463	0.0000
16	fixed	<NA>	yards_endzone	0.0001	0.0005	0.1844	30687.7872	0.8537
17	ran_pars	rusher_id	sd__(Intercept)	0.4088	NA	NA	NA	NA
18	ran_pars	blocker_id	sd__(Intercept)	0.2121	NA	NA	NA	NA
19	ran_pars	defensive_team	sd__(Intercept)	0.2477	NA	NA	NA	NA
20	ran_pars	offensive_team	sd__(Intercept)	0.2555	NA	NA	NA	NA
21	ran_pars	Residual	sd__Observation	2.0594	NA	NA	NA	NA

ICC

- ▶ ICC close to 0: responses are more independent, multilevel model structure is not as relevant
- ▶ ICC close to 1: repeated observations provide no new information, multilevel group structure is important

```
# this matches Table 7 in arxiv.org/pdf/2305.10262.pdf
library(merTools)
strain_lmer |>
  VarCorr() |>
  as_tibble() |>
  mutate(icc = vcov / sum(vcov))
```

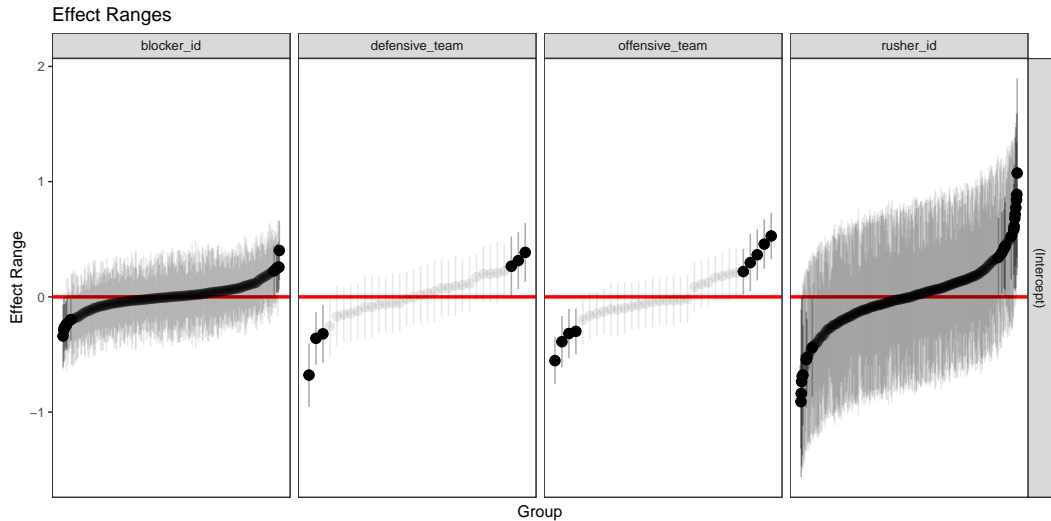
```
# A tibble: 5 x 6
```

	grp	var1	var2	vcov	sdcor	icc
	<chr>	<chr>	<chr>	<dbl>	<dbl>	<dbl>
1	rusher_id	(Intercept)	<NA>	0.167	0.409	0.0365
2	blocker_id	(Intercept)	<NA>	0.0450	0.212	0.00982
3	defensive_team	(Intercept)	<NA>	0.0614	0.248	0.0134
4	offensive_team	(Intercept)	<NA>	0.0653	0.256	0.0143
5	Residual	<NA>	<NA>	4.24	2.06	0.926

Group-level effects

```
# cran.r-project.org/web/packages/merTools/vignettes/Using_predictInterval.html  
strain_lmer |>  
  REsim() |>  
  plotREsim()
```

Group-level effects



References

Nguyen, Q., Yurko, R., & Matthews, G. J. (2024). Here Comes the STRAIN: Analyzing Defensive Pass Rush in American Football with Player Tracking Data. *The American Statistician*, 78(2), 199–208. <https://doi.org/10.1080/00031305.2023.2242442>

Extended edition: <https://arxiv.org/pdf/2305.10262.pdf>

GitHub: <https://github.com/getstrained/intro-strain>