Hierarchical models

Jay Brophy MD PhD
Departments of Medicine, Epidemiology and Biostatistics, McGill University

2025-11-03

Conflicts of interest

No conflicts of interest
To the best of my knowledge I'm equally disliked, or at best deemed irrelevant, by all drug and device companies

AI generated image

Objectives

Review of probability concepts

Review of hierarchical modeling concepts

Realization of their practical applications

Hierarchical models (models with memory)

Data often not **iid** (e.g. repeat or clustered measurements) and must account for the dependency structure

Examples include:
- nested data (patients within hospitals, within regions, with countries)
- longitudinal (panel) studies or time-series cross-sectional data
- meta-analysis

Simple averaging can be dangerous leading to false confidence e.g. Simpson's paradox

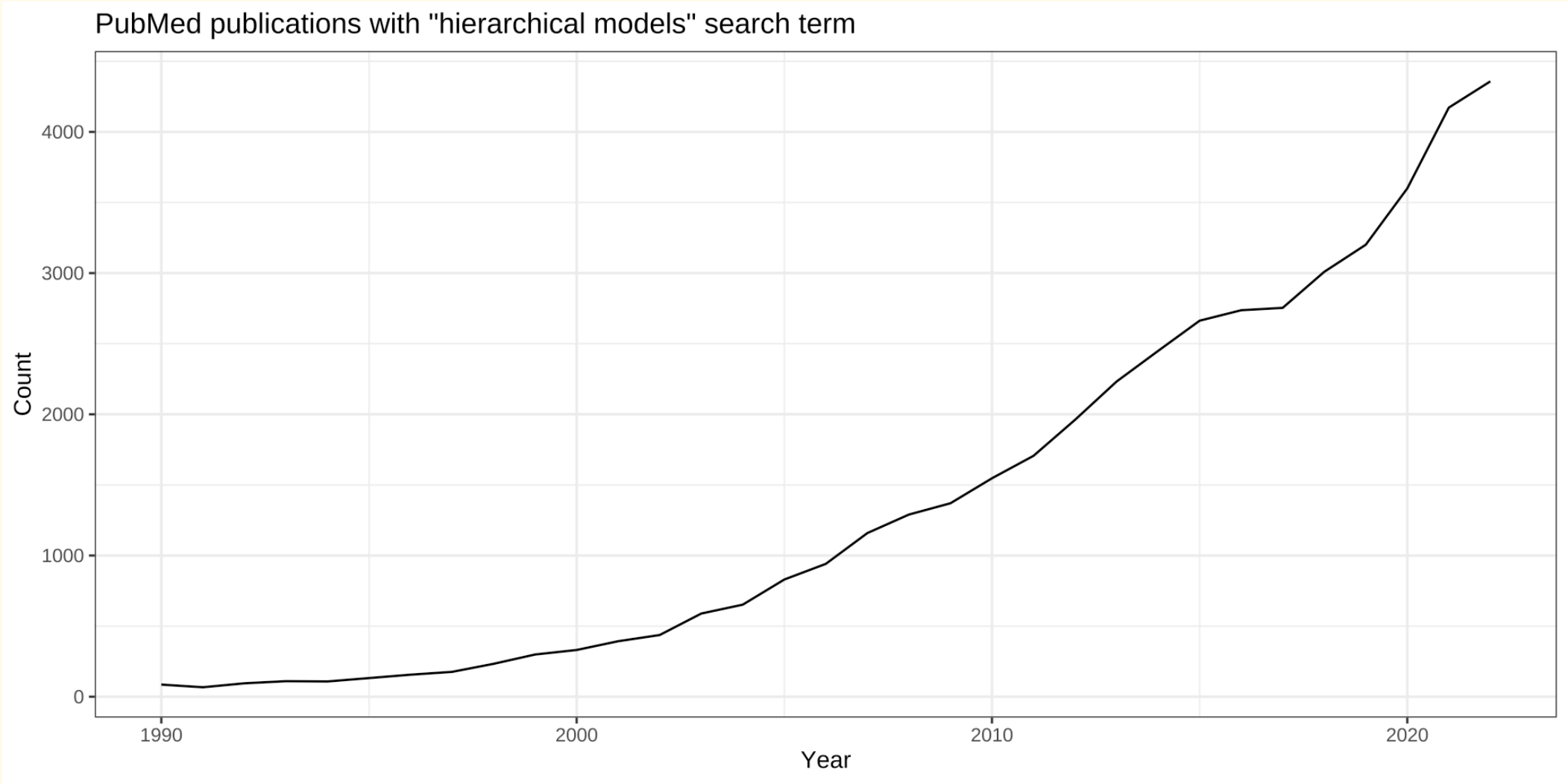
Hierarchical models (benefits)

Benefits of the multilevel approach

(1) Provide robust and conservative estimates across groups
(2) Estimate variation on all groups of the model
(3) Predict new values, even for groups not originally present
(4) Model complex data structures (e.g., nested, crossed)
(5) Avoid multiple comparisons problems
(6) Allow for partial pooling of information across groups

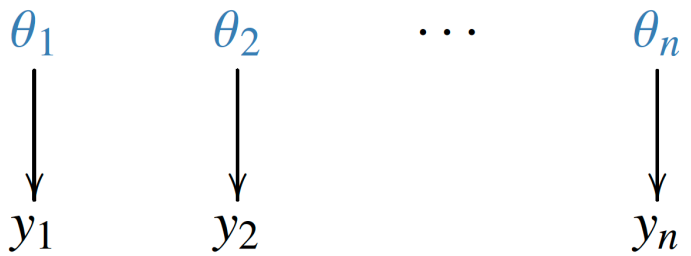
Multilevel models allow us to preserve all uncertainties & **should be default models**

Increasing use of hierarchical models

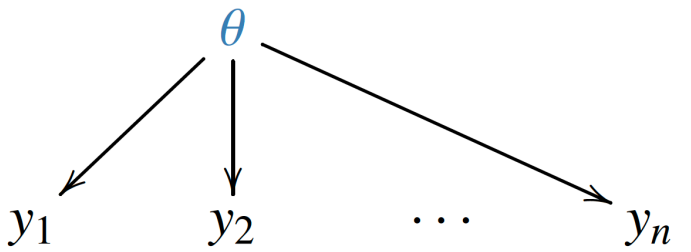


Model comparisons

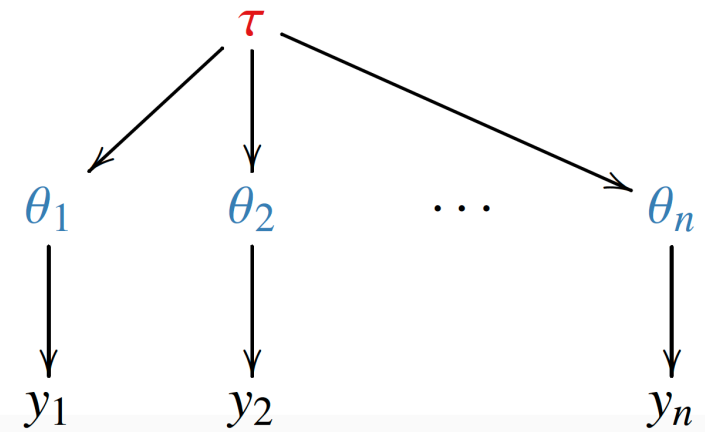
- "Separate model" (model with separate/independent effects)



- "Joint model" (model with a common effect / pooled model)



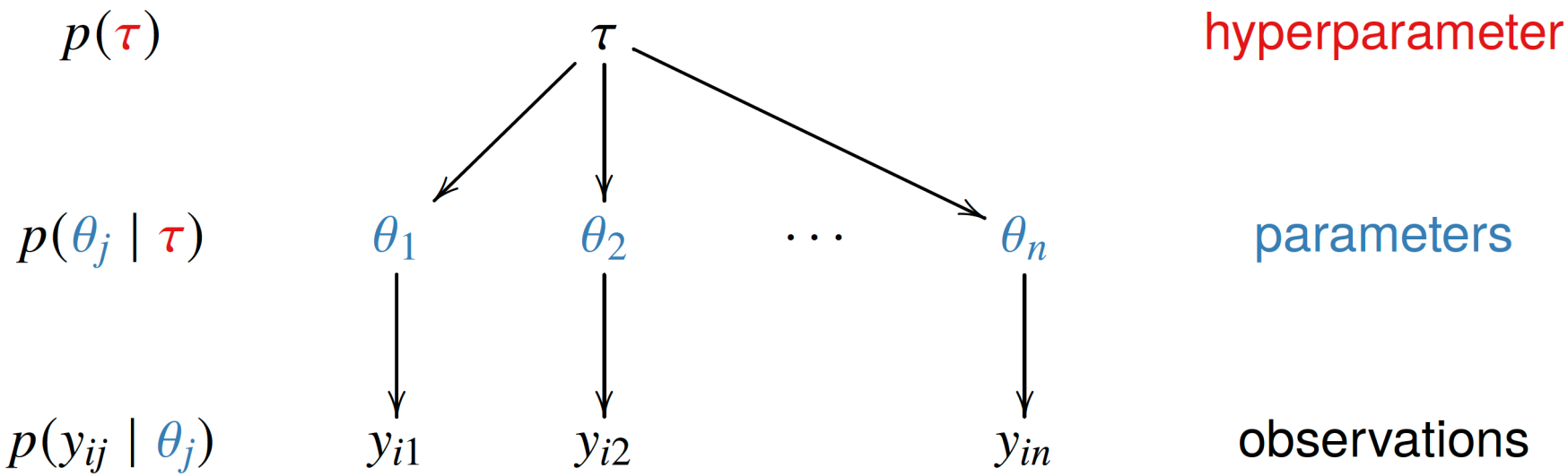
- Hierarchical model



Hierarchical model: terms

Level 1: observations given parameters $p(y_{ij} \mid \theta_j)$

Level 2: parameters given hyperparameters $p(\theta_j \mid \tau)$

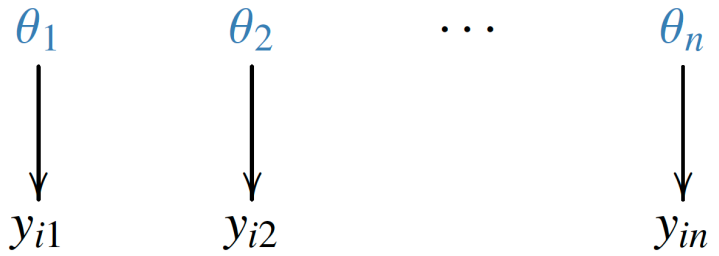


Joint posterior

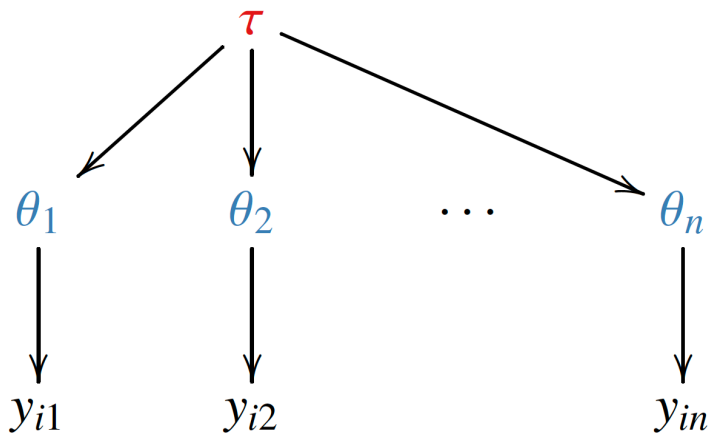
$$p(\theta, \tau \mid y) \propto p(y \mid \theta, \tau)p(\theta, \tau)$$
$$\propto p(y \mid \theta)p(\theta \mid \tau)p(\tau)$$

Hierarchical model example

- Example: CVD treatment effectiveness
  - in hospital j the survival probability is θ_j
  - observations y_{ij} tell whether patient i survived in hospital j



  - sensible to assume that θ_j are similar



  - natural to think that θ_j have common population distribution
  - θ_j is not directly observed and the population distribution is unknown

Hierarchical model: meta-analysis

Bayesian random effects meta-analysis model follows the same hierarchical approach
Level 1

$y_j = \theta_j + \epsilon$ where error term $\epsilon_j \sim normal(0, s_j)$ models the uncertainty of the effect from each individual study, θ_j.
Assumes estimate y_j from the j-th study is unbiased and normally distributed with standard error s_j.

Level 2

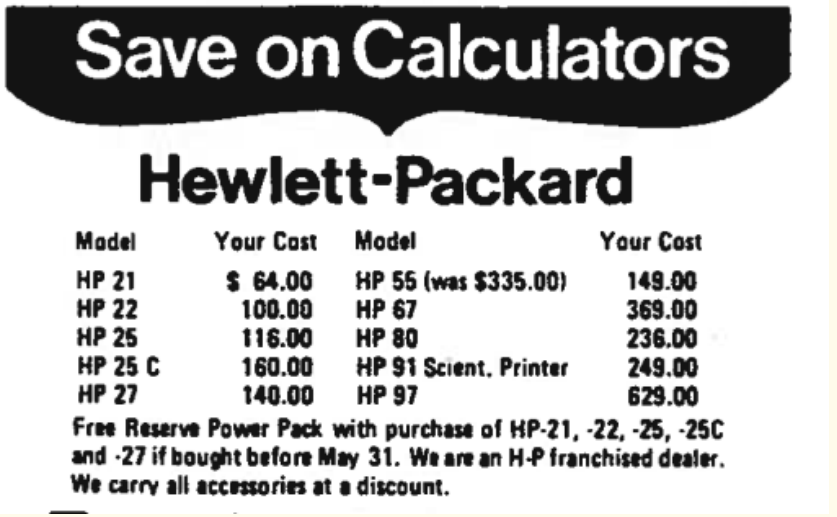
$\theta_j = \mu + \mu_j$ where error term $\mu_j \sim normal(0, \tau)$
Assumes μ estimates the average treatment effect in the (hypothetical) superpopulation of similar studies, θ_j is the effect in the j-th individual study and the variance τ^2 is referred to as the heterogeneity

Stein's paradox (1997)

Stein's Paradox in Statistics

The best guess about the future is usually obtained by computing the average of past events. Stein's paradox defines circumstances in which there are estimators better than the arithmetic average

by Bradley Efron and Carl Morris



Save on Calculators

Hewlett-Packard

Model	Your Cost	Model	Your Cost
HP 21	$ 64.00	HP 55 (was $335.00)	149.00
HP 22	100.00	HP 67	369.00
HP 25	116.00	HP 80	236.00
HP 25 C	160.00	HP 91 Scient. Printer	249.00
HP 27	140.00	HP 97	629.00

Free Reserve Power Pack with purchase of HP-21, -22, -25, -25C and -27 if bought before May 31. We are an H-P franchised dealer. We carry all accessories at a discount.

Baseball

Early application of hierarchical models in statistics

Calculating each player's batting average based on early season performance to predict end-of-season performance

Which model to predict final average?

3 possible predictive models

(a) complete pooling, which assumes each player is the same, $\boldsymbol{y}$, corresponds to zero population variance
(b) no pooling, which assumes the players are unrelated, y_i, corresponds to infinite population variance
(c) partial pooling (hierarchical), where each player is assumed to have a different chance of success, but the data for all of the observed players informs the estimates for each individual player

Pooling = f(variance in the population), more variance -> less pooling
Stein's paradox: New Stein/James (partial) estimators, $(z_i = \boldsymbol{y} + c(y_i - \boldsymbol{y}))$, predict better than individual averages, y_i, or the global average, $\boldsymbol{y}$ where **c** is the shrinkage factor

Stein's paradox (1977)

If > 3 means, estimating each by its own mean is **not** the most efficient

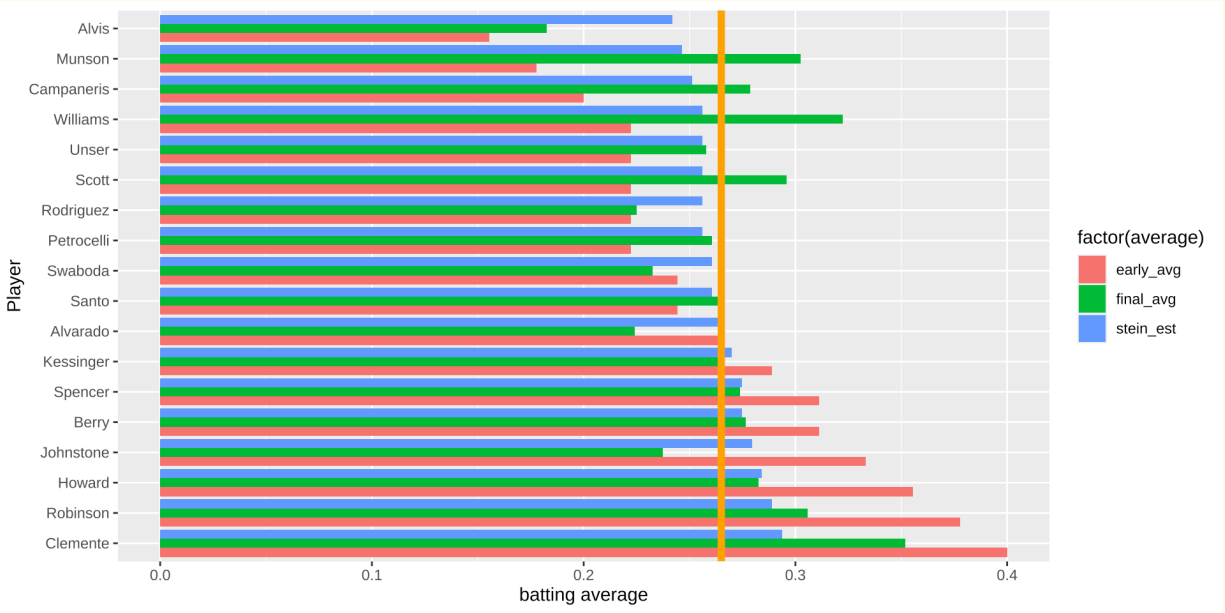
Essence of the Stein method is that of "borrowing" of information or "shrinkage" will produce "better" estimates, meaning lowered mean squared errors

Baseball data for early season average (hits (successes) / 45 chances (AB)) for 18 players

```r
library(rstanarm)
data(bball1970)
bball <- bball1970
# graph on next slide
bball_graph <- bball %>%
  mutate(final_avg = (Hits + RemainingHits) / (AB + Remaining
         stein_est = .265 + .212*(early_avg-.265)) %>%
  dplyr::select(Player, final_avg, early_avg, stein_est) %>%
  pivot_longer(!Player, names_to = "average", values_to = "co
  ggplot(aes(x=Player, y=count, fill=factor(average)))+
  ylab("batting average") +
  geom_col(position='dodge') +
  geom_hline(yintercept = .265, linewidth=2, color = "orange"
#  geom_hline(yintercept = .269, linewidth=2, color = "black"
  coord_flip()
```

	Player	AB	Hits	RemainingAB	RemainingHits
1	Clemente	45	18	367	127
2	Robinson	45	17	426	127
3	Howard	45	16	521	144
4	Johnstone	45	15	275	61
5	Berry	45	14	418	114
6	Spencer	45	14	466	126
7	Kessinger	45	13	586	155
8	Alvarado	45	12	138	29
9	Santo	45	11	510	137
10	Swaboda	45	11	200	46
11	Petrocelli	45	10	538	142
12	Rodriguez	45	10	186	42
13	Scott	45	10	435	132
14	Unser	45	10	277	73
15	Williams	45	10	591	195
16	Campaneris	45	9	558	159
17	Munson	45	8	408	129
18	Alvis	45	7	70	14

Which average is better?

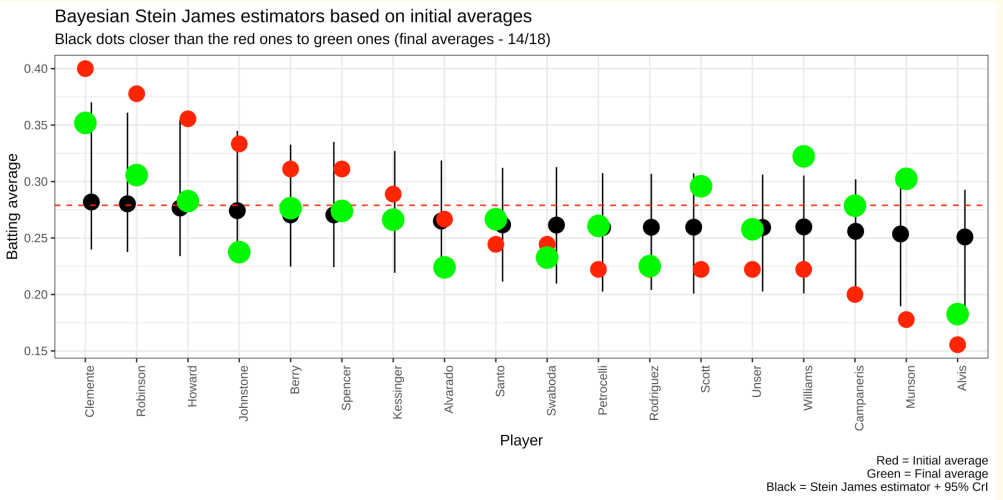


Key points

1. Stein estimators (blue bar) closer to early global mean than individual early averages (red bar) (“shrinkage”)
2. Shrinkage is greatest for points farthest from the early global mean
3. Stein estimators (blue bar) closer to individual final avg (green bar) 14 vs 4 times than for the individual early avg (red bar) (i.e beats separate model)
4. Stein estimators (blue bar) closer to individual final avg (green bar) 13 vs 5 times than for the early global mean (i.e beats pooled model)

Bayesian approach

Stein estimator is not Bayesian as it is not dependent on priors, but it is a precursor to Bayesian multilevel models (shrinkage, borrowing information)
Full Bayesian analysis (with weakly informative priors) is possible and follows using `rstanarm` package




```
SEED <- 101
fit_partialpool <- stan_glmer(cbind(Hits, AB - Hits) ~ (1 | P
    family = binomial("logit"), seed = SEED, refresh=0)
# shift each player's estimate by intercept (and then drop in
shift_draws <- function(draws) {sweep(draws[, -1], MARGIN = 1
summary_stats <- function(posterior) {x <- invlogit(posterior
  t(apply(x, 2, quantile, probs = c(0.05, 0.5, 0.95))) }
alphas <- shift_draws(as.matrix(fit_partialpool));partialpool
partialpool <- partialpool[-nrow(partialpool),];rownames(part
batting_avg <- function(x) print(format(round(x, digits = 3),
```

James–Stein estimators -> implications on ML era - Efron 2023

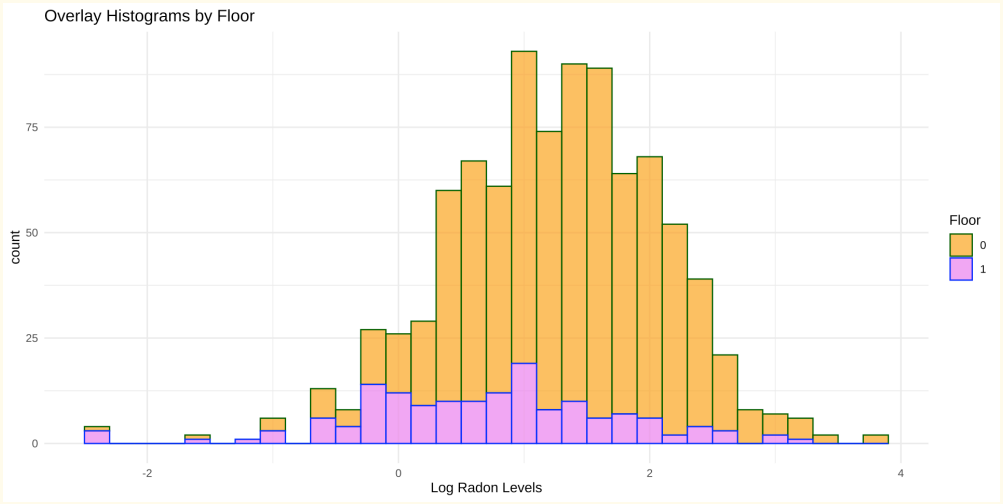
Epidemiology / Public Health Example - Radon

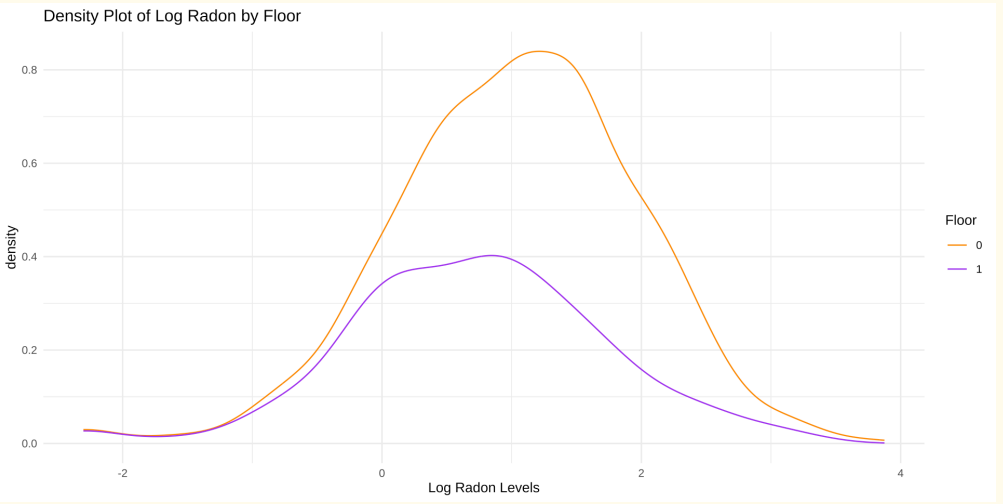
Radon, radioactive gas, highest cause of lung cancer in non-smokers.
Believed to enter houses through the basement
Concentration differ regionally due to different soil types

919 obs. of 4 variables from houses in 85 counties in Minnesota
- log_radon - - **log_radon**: Radon measurement from the house (log scale)
- **floor**: 0 = basement, 1 = first floor
- **county**: County name (factor)
- **log_uranium**: Uranium level in the county (log scale)

```
# Load required libraries
pacman::p_load(tidyverse, cmdstanr, ggplot2, ggthemes, rstanarm, brms)
xlabels_90 <- theme(axis.text.x = element_text(angle = 90, hjust = 1))
mn_radon <- read_csv("data/mn_radon.csv", show_col_types = FALSE) # Read data
```

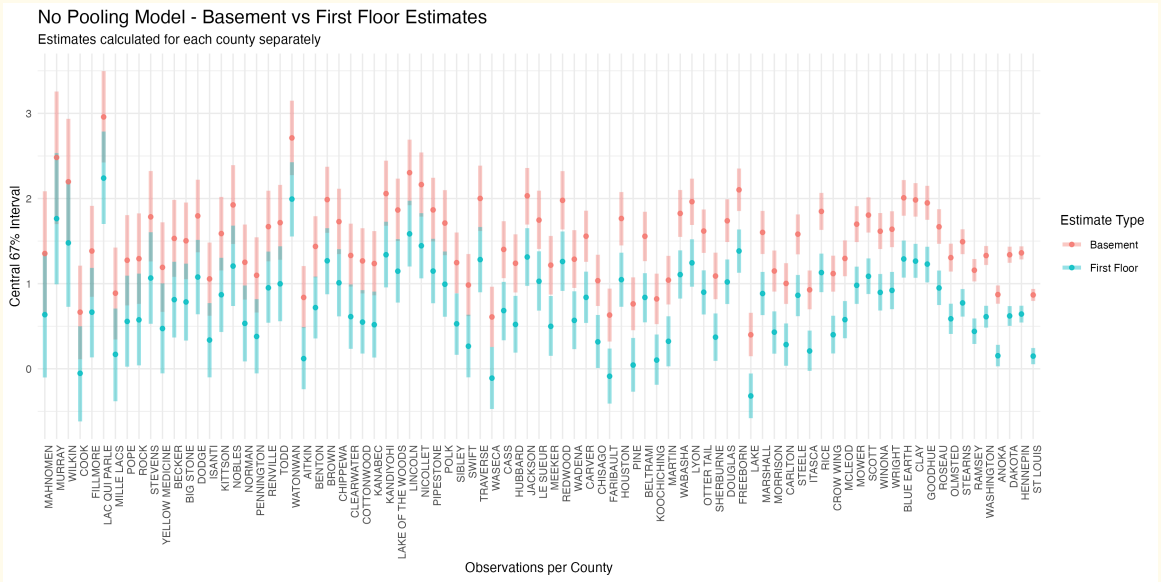
Radon Exploratory Data Analysis





Radon models - no pooling

- No pooling; assumes counties are completely different
- Assumes nothing to be gained from borrowing information across groups
- Builds separate models for each county



Math speak $$radon_{i,c} = \alpha_c + \beta_c * floor_{i,c} + \epsilon_c$$

Where *i* represents the measurement and *c* the county
IOW each county *c* has its own intercept α and slope β

Radon models - complete pooling

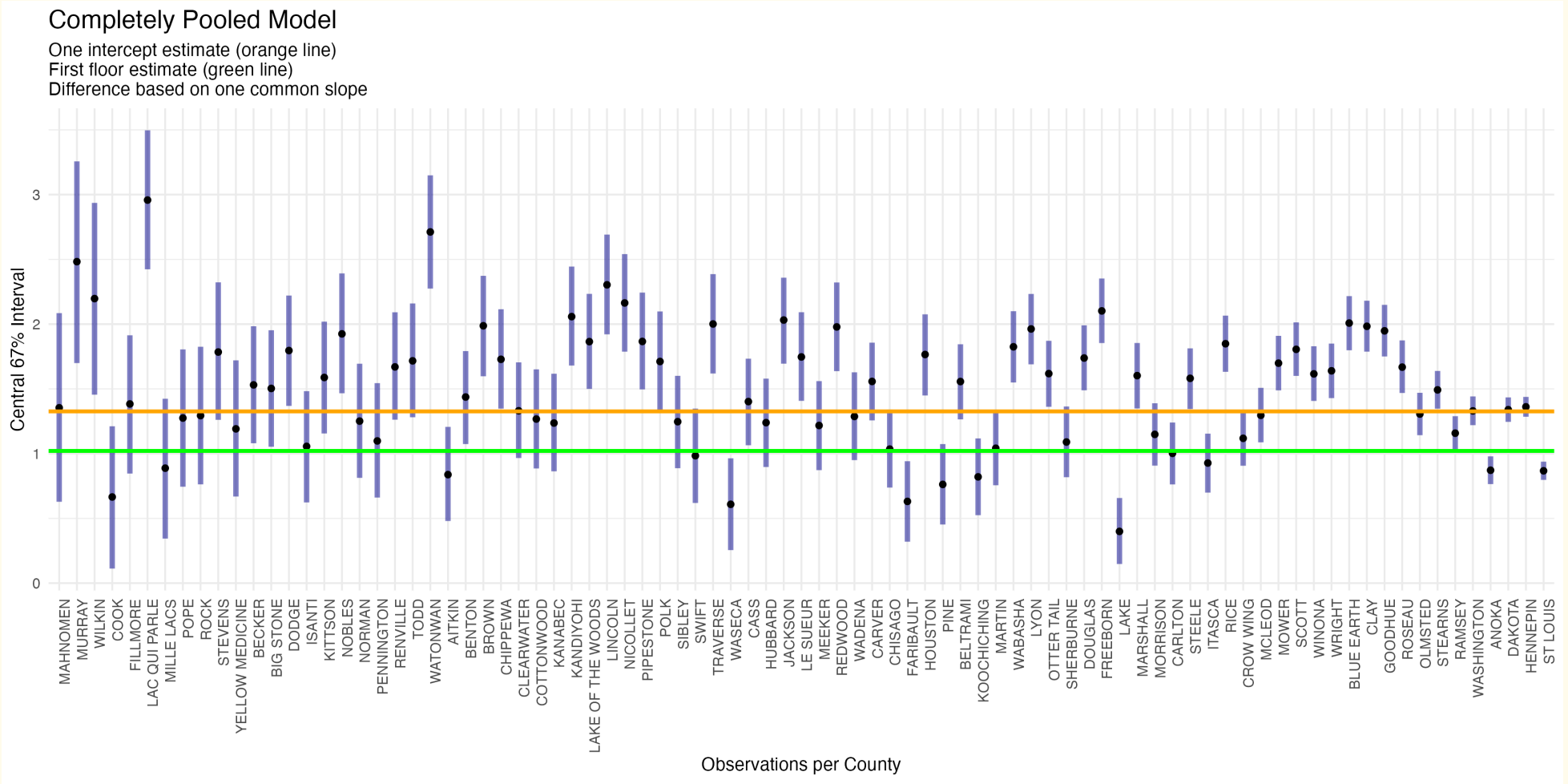
Pooled model, `log_radon ~ floor`, ignores any grouping structure (e.g., counties), assumes observations are iid, `lm()` sufficient provided no priors to include

Straightforward interpretation `floor` coefficient = average (uniform) effect across all observations, ignores any variation between different counties.

Provides overall average effect but can lead to biased or misleading conclusions if significant group-level variability exists, underestimates the standard errors of the estimated effects by ignoring intra-group correlation.

Radon models - complete pooling

- Pool into one big regression model with 1 interceptand 1 slope, δ between orange & green lines
- Assumes all counties are identical
- Opposite extreme to no pooling



Math speak $radon_i = \alpha + \beta * floor_i + \epsilon$

Radon models - partial pooling

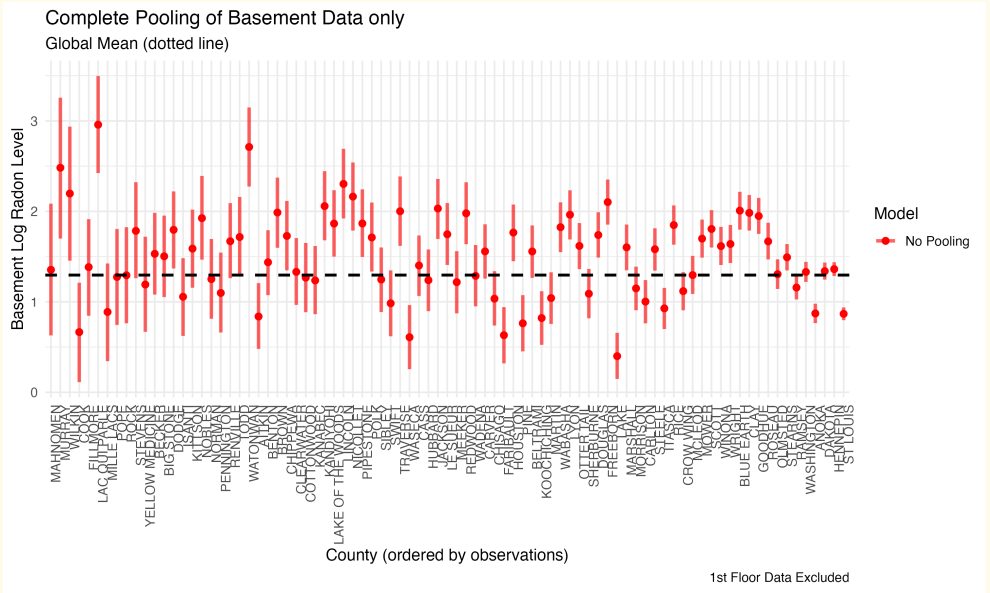
Hierarchical, AKA mixed-effects, model, `log_radon ~ floor + (1|county_id)`,intercept varies by county around the overall average radon level

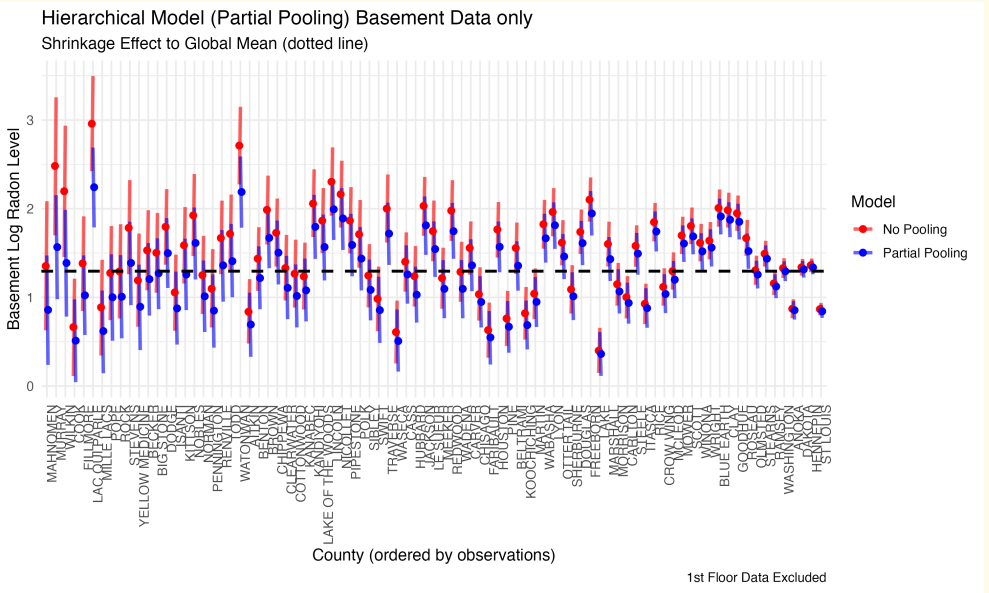
`log_radon ~ floor + (1 + floor | county_id)`, both intercept and slope vary by county around the overall average radon level and floor effect

Balances between pooling and separation

If completely pooled & hierarchical models are very different, -> high between-group variation, more complex hierarchical model is likely better.

Radon models - complete vs. partial pooling plots





Math speak $\alpha_c \sim \mathrm{N}(\mu_\alpha, \sigma_\alpha^2)$

1. More movement of point estimate toward global mean with
    i. “trials” farther away from global mean
    ii. smaller sample size
2. More shrinkage of credible intervals with smaller studies

Radon Bayesian models

Easiest to use front end packages like `rstanarm` and `brms`, especially designed for Bayesian hierarchical analyses

These packages are not necessary for no pooling case (provided no prior info to include) as `lm()` or `glm()` sufficient

```
library(brms)
library(loo)
# ====== Complete Pooling ======
model_cp_brms <- brm(log_radon ~ floor, data = mn_radon, fami

# ====== Partial Pooling ======
model_pp_brms <- brm(log_radon ~ floor + (1|county_id), data

brms_comp <- loo_compare(loo(model_cp_brms), loo(model_pp_brm
save(brms_comp, file="output/brms_comp.RData")
```

```
library(rstanarm)
library(loo)
mn_radon <- read_csv("data/mn_radon.csv", show_col_types = FA

# ====== Complete Pooling ======
model_cp_rstanarm <- stan_glm(log_radon ~ floor, data = mn_ra

# ====== Partial Pooling ======
model_pp_rstanarm <- stan_glmer(log_radon ~ floor + (1|county

save(model_pp_rstanarm, file="output/model_pp_rstanarm.RData'
rstan_comp <- loo_compare(loo(model_cp_rstanarm), loo(model_p
save(rstan_comp, file="output/rstan_comp.RData")
```

Radon models - partial pooling model

These front end packages use Stan code, not exactly trivial to write
But can be visualized with `brms::stancode(model_pp_brms)`

```
# Example Stan code:
data {
  int<lower=1> N;  // observations
  int<lower=1> J;  // counties
  array[N] int<lower=1, upper=J> county;
  vector[N] x;     // floor
  vector[N] y;     // radon
}
parameters {
  vector[J] alpha;
  real beta;
  real<lower=0> sigma;
}
model {
  // Model specifications
  alpha ~ normal(0, 10);
  beta ~ normal(0, 10);
  sigma ~ normal(0, 10);

  // Likelihood
  y ~ normal(alpha[county] + beta * x, sigma);
}
generated quantities {
  array[N] real log_lik;     // Log-likelihood for each observation
  array[N] real y_rep;       // Posterior predictive checks
  for (i in 1:N) {
    log_lik[i] = normal_lpdf(y[i] | alpha[county[i]] + beta * x[i], sigma);
    y_rep[i] = normal_rng(alpha[county[i]] + beta * x[i], sigma);  // Generate predictive samples
  }
```

Radon models - which is better?

```
load("rstan_comp.RData")
rstan_comp
```

```
                  elpd_diff se_diff
model_pp_rstanarm   0.0       0.0
model_cp_rstanarm -53.2      10.6
```

```
load("brms_comp.RData")
brms_comp
```

```
                elpd_diff se_diff
model_pp_brms   0.0       0.0
model_cp_brms -52.8      10.6
```

Explanation of elpd_diff

Δ expected log pointwise predictive density (epld) between two statistical models measures model predictive performance

Metric used with the `loo` (leave-one-out cross-validation) package when comparing Bayesian models with higher values -> better predictive performance.

elpd_diff = -53.2 (52.8) suggests that the first model (hierarchical) likely to be a better predictor of new data than completely pooled model with $z \approx 5$

Interpretation: Strong evidence that a hierarchical model is preferred over the completely pooled model.

Radon models

What about other possible confounders e.g. `log_uranium`

```
# The subset of the radon data we need is included in rstanarm
library("rstanarm"); data(radon)
# Now adjust the pooled  model for the county-level uranium predictor
modelA <- stan_lmer(
 log_radon ~ floor + log_uranium + floor:log_uranium + (1 + floor | county),
  data = radon, cores = 4, iter = 2000, chains = 4
)
saveRDS(modelA, file = "output/modelA.rds")
# Recall the hierarchical model without the county-level uranium predictor
modelB <- update(modelA, formula = log_radon ~ floor + (1 + floor | county))
saveRDS(modelB, file = "output/modelB.rds")
```

```
# loading outcome to save repeating MCMC calculations
modelA <- readRDS("output/modelA.rds");modelB <- readRDS("output/modelB.rds")
# Compare models needs to separate code block from preceeding (why?)
looA <- loo(modelA); looB <- loo(modelB)
print(loo_compare(looA, looB))
```

```
       elpd_diff se_diff
```

modelA 0.0 0.0
modelB -9.8 5.1

elpd_diff = -9.8 suggests that the including the variable `log_uranium` in the hierarchical likely leads to better predictor compared to the baseline hierarchical model without it, but se = 5.1 suggesting some uncertainty (z = - 1.92)

Interpretation: Although there is some uncertainty does suggest hierarchical model including county uranium levels (model A) is preferred.

Radon output table

```
1 head(summary(modelA))
```

```
                                    mean         mcse         sd        10%
(Intercept)                   1.49961323 0.0005561482 0.03486938  1.4554032
floor                        -0.65714227 0.0010488821 0.07912519 -0.7557767
log_uranium                   0.78698453 0.0014784930 0.09189782  0.6686785
floor:log_uranium            -0.43203819 0.0027913042 0.20460791 -0.6922567
b[(Intercept) county:AITKIN] -0.01533251 0.0015191119 0.12560828 -0.1696316
b[floor county:AITKIN]        0.01564660 0.0028778328 0.21529226 -0.2296711
                                     50%        90% n_eff      Rhat
(Intercept)                   1.499581802  1.5436340  3931 1.0004394
floor                        -0.659742443 -0.5542719  5691 0.9992758
log_uranium                   0.789064497  0.9040145  3863 0.9997451
floor:log_uranium            -0.431594916 -0.1716113  5373 0.9998638
b[(Intercept) county:AITKIN] -0.011744665  0.1349220  6837 0.9995214
b[floor county:AITKIN]        0.005217733  0.2764645  5597 0.9998745
```

The effect of floor is therefore $exp^{-0.66}$ = 0.51 decrease in radon measurements from basement to first floor

Multinational RCT

PLATO NEJM 2009 multinational RCT of antiplatelets, ticagrelor vs. clopidogrel > 5,000 citations

Results Primary end point — composite of death from vascular causes, myocardial infarction, or stroke — occurred in 9.8% of ticagrelor subjects vs. 11.7% of clopidogrel subjects (HR, 0.84; 95% CI 0.77 to 0.92; P<0.001)
Conculsion Ticagrelor significantly reduced the primary outcome
Data - 862 centers, 43 countries, 6 geographic regions (iid?)?

```
1 library(epiR)
2 dat <- read.csv("data/Plato_FDA_470.csv", header = TRUE)
3 kable(dat)
```

Region	tic_N	tic_O	clo_N	clo_O
US	707	84	706	67
Eastern Europe	3820	299	3825	394
Western Europe	2725	240	2704	281
Asia	819	90	812	114
Latin America	621	91	615	104
Other	641	60	628	54

Reproduce NEJM results

NEJM -> time to event analysis. With only aggregate data, limited to cumulative RR (`epiR::epi.2by2`)

```
nt <- sum(dat$tic_N); nc <- sum(dat$clo_N); et <- sum(dat$tic_O); ec <- sum(dat$clo_O)
total <- matrix(c(et,nt-et,ec,nc-ec), nrow = 2, byrow = TRUE, dimnames = list(c("Ticagrelor", "Clopidogrel"), c("Died", "Alive"
epi.2by2(dat = as.table(total), method = "cohort.count", conf.level = 0.95, units = 100, outcome = "as.columns")
```

```
            Outcome+    Outcome-      Total                Inc risk *
Exposure+         864        8469       9333        9.26 (8.68 to 9.86)
Exposure-        1014        8276       9290      10.91 (10.29 to 11.57)
Total            1878       16745      18623       10.08 (9.66 to 10.53)

Point estimates and 95% CIs:
-------------------------------------------------------------------
Inc risk ratio                                 0.85 (0.78, 0.92)
Inc odds ratio                                 0.83 (0.76, 0.92)
Attrib risk in the exposed *                  -1.66 (-2.52, -0.79)
Attrib fraction in the exposed (%)           -17.90 (-28.50, -8.18)
Attrib risk in the population *               -0.83 (-1.60, -0.06)
Attrib fraction in the population (%)        -8.24 (-9.89, -6.55)
-------------------------------------------------------------------
Uncorrected chi2 test that OR = 1: chi2(1) = 14.106 Pr>chi2 = <0.001
Fisher exact test that OR = 1: Pr>chi2 = <0.001
 Wald confidence limits
 CI: confidence interval
 * Outcomes per 100 population units
```

Results concordant with reported (HR, 0.84; 95% CI 0.77 to 0.92; P<0.001)

Data clustering

Original treated observations as **iid**, ignores possible clustering within specific geographic regions -> error underestimation -> over confidence in results precision

US data

```
# to get individual non-pooled results for each region
dat2 <- dat[dat$Region=="US",]
mat <- matrix(c(dat2$tic_0, dat2$tic_N - dat2$tic_0, dat2$clo
              dimnames = list(c("Ticagrelor", "Clopidogrel"),
tt <- epi.2by2(dat = as.table(mat), method = "cohort.count",
tt$massoc.summary[1,1:4]
```

```
            var      est     lower    upper
1 Inc risk ratio 1.251958 0.9241111 1.696115
```

Eastern Europe data

```
# to get individual non-pooled results for each region
dat2 <- dat[dat$Region=="Eastern Europe",]
mat <- matrix(c(dat2$tic_0, dat2$tic_N - dat2$tic_0, dat2$clo
              dimnames = list(c("Ticagrelor", "Clopidogrel"),
tt <- epi.2by2(dat = as.table(mat), method = "cohort.count",
tt$massoc.summary[1,1:4]
```

```
            var       est     lower     upper
1 Inc risk ratio 0.7598766 0.6583074 0.8771167
```

Are these differences statistically significant?

Data clustering

Are these differences statistically significant?
Quick R function to test interaction btw 2 RR using their CIs

```r
# Tests for interaction between two risk ratios using their confidence intervals.
test_interaction <- function(risk_1, ci_1, risk_2, ci_2, alpha = 0.05) {
  # Calculate the log-odds (log risk ratios) & variances for 1 and 2
  log_odds_1 <- log(risk_1); log_odds_2 <- log(risk_2)
  var_1 <- ((log(ci_1[2]) - log(ci_1[1])) / (2 * 1.96))^2; var_2 <- ((log(ci_2[2]) - log(ci_2[1])) / (2 * 1.96))^2
  # Compute the difference in log-odds (interaction effect) & variance
  diff_log_odds <- log_odds_2 - log_odds_1; var_diff <- var_1 + var_2
  # Compute the z-statistic, p-value
  z_stat <- diff_log_odds / sqrt(var_diff); p_value <- 2 * (1 - pnorm(abs(z_stat)))
  risk_ratio_interaction <- exp(diff_log_odds) # Convert back to risk ratio & 95% CI
  lower_ci <- exp(diff_log_odds - 1.96 * sqrt(var_diff))
  upper_ci <- exp(diff_log_odds + 1.96 * sqrt(var_diff))
  significant <- ifelse(p_value < alpha, "Significant", "Not Significant") # Check significance
  return(list(z_statistic = z_stat, p_value = p_value,
    significant = significant,
    risk_ratio_interaction = risk_ratio_interaction,
    ci_interaction = c(lower_ci, upper_ci)
  ))
}

result <- test_interaction(0.75, c(0.66,0.88), 1.25, c(0.92,1.69))
print(result)
```

```
$z_statistic
[1] 2.976605

$p_value
[1] 0.002914594

$significant
[1] "Significant"

$risk_ratio_interaction
[1] 1.666667
```

```
$ci_interaction
[1] 1.190607 2.333077
```

Different models

Possible solution - hierarchical modeling approach to account for clustering by region
Separate regions (like Bayesian MA of RCTs), can be modeled as follows y_j = log(RR)

$$y_j \sim N(\theta_j, \sigma_j^2)$$
$$\theta_j \sim N(\mu, \tau)$$

where $y_j = log(RR_j)$ and σ_j^2 is assumed to be known with certainty (large sample sizes)
Meta-analyses are typically concerned with the overall mean, μ

Recall, in general, three ways to estimate θ_j;
1. No-pooling: there is a separate model for each study and $\theta_j = y_j$. i.e. hierarchical model in which the between study variation $\tau = \infty$
2. Complete-pooling: patients in each study are random samples from a common distribution so $\theta_j = \mu$. i.e. hierarchical model in with $\tau = 0$
3. Partial-pooling: a compromise between 1. and 2. In this case τ is estimated from the data and θ_j is closer to μ when τ is small relative to σ_j^2 and closer to y_j when the reverse is true

Meta-analysis

Random effects meta-analysis is a form of hierarchical modeling (regression analysis) where both within and between group variations are considered

A relevant statistic is the relative risk ratio, or p_{1j}/p_{0j}, where $p_{1j} = d_{1j}/n_{1j}$ and $p_{0j} = d_{0j}/n_{0j}$
Log relative risk ratio, $y_j = log(p_{1j}) - log(p_{0j})$ preferred, since $\approx N(\mu, \sigma)$
Variance of each y_j calculated by treating p_{1j} and p_{0j} as sample proportions and using the delta method, so y_j variance is

$$\sigma_j^2 \approx \frac{1 - p_{1j}}{n_{1j} p_{1j}} + \frac{1 - p_{0j}}{n_{0j} p_{0j}}$$

```
# hand calculation of rr and se
dat$p1 <- dat$tic_0/dat$tic_N; dat$p0 <- dat$clo_0/dat$clo_N; dat$rr <- dat$p1/dat$p0; dat$lrr <- log(dat$rr)
dat$lse <- sqrt((1 - dat$p1)/(dat$p1 * dat$tic_N) + (1 - dat$p0)/(dat$p0 * dat$clo_N)); dat$lower <- exp(dat$lrr - qnorm(.975)
# easier would be
# dat <- metafor::escalc(measure="RR", ai=tic_0, n1i = tic_N, ci=clo_0,  n2i =clo_N, data=dat)
```

Rather than doing this by hand, `escalc` from `metafor` package also works well

Meta-analysis - Tabular data

Based on hand calculations from previous slide

```
dat
```

```
          Region tic_N tic_O clo_N clo_O         p1         p0        rr
1             US   707    84   706    67 0.11881188 0.09490085 1.2519580
2 Eastern Europe  3820   299  3825   394 0.07827225 0.10300654 0.7598766
3 Western Europe  2725   240  2704   281 0.08807339 0.10392012 0.8475105
4           Asia   819    90   812   114 0.10989011 0.14039409 0.7827260
5  Latin America   621    91   615   104 0.14653784 0.16910569 0.8665459
6          Other   641    60   628    54 0.09360374 0.08598726 1.0885769
           lrr        lse     lower     upper
1  0.22470875 0.15491700 0.9241111 1.6961153
2 -0.27459929 0.07320748 0.6583074 0.8771167
3 -0.16545202 0.08359779 0.7194267 0.9983979
4 -0.24497252 0.13202467 0.6042688 1.0138866
5 -0.14324021 0.13178799 0.6692886 1.1219402
6  0.08487123 0.17897700 0.7665026 1.5459826
```

where $lrr = log(RR_i)$

e.g. for US lrr = log((84/707)/(67/706)) = 0.2247
US RR = exp(.2247) = 1.25

Meta-analysis tabular results

Quantitative results from packages `metafor::rma` or `meta::metabin`

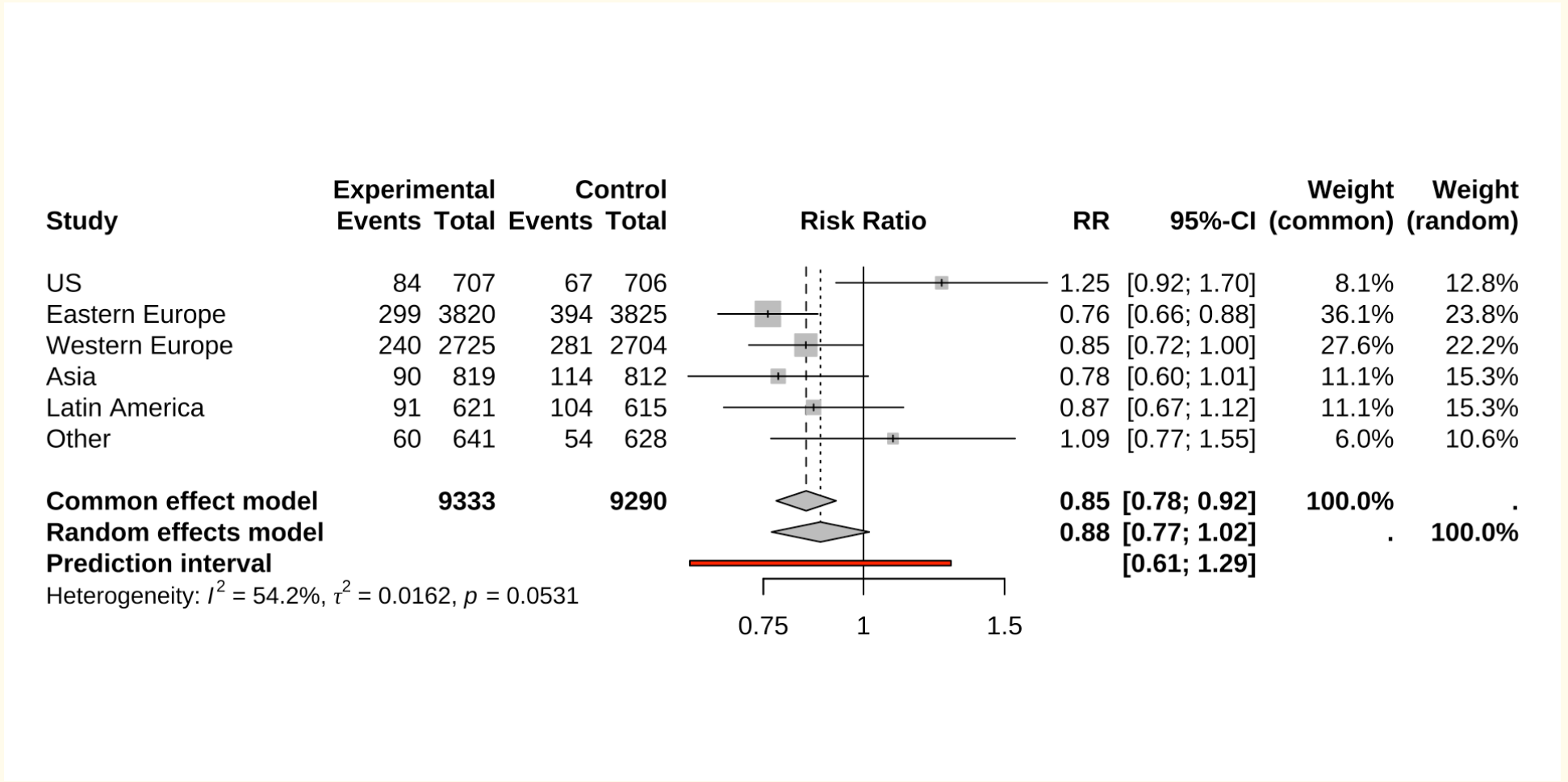
```r
library(metafor)
me.fe <- rma(dat$lrr, sei=dat$lse, method = "FE") # exponentiation c(exp(me.fe$b), exp(me.fe$ci.lb), exp(me.fe$ci.ub))
me.re <- rma(dat$lrr, sei=dat$lse, method = "REML") # exponentiation c(exp(me.re$b), exp(me.re$ci.lb), exp(me.re$ci.ub))
plot_Plato <- meta::metabin(dat$tic_O, dat$tic_N, dat$clo_O, dat$clo_N, sm="RR", method ="I", studlab=dat$Region, prediction=TR
print(summary(plot_Plato,prediction=TRUE), digits=2)
```

```
                 RR       95%-CI %W(common) %W(random)
US             1.25 [0.92; 1.70]        8.1       12.8
Eastern Europe 0.76 [0.66; 0.88]       36.1       23.8
Western Europe 0.85 [0.72; 1.00]       27.6       22.2
Asia           0.78 [0.60; 1.01]       11.1       15.3
Latin America  0.87 [0.67; 1.12]       11.1       15.3
Other          1.09 [0.77; 1.55]        6.0       10.6

Number of studies: k = 6
Number of observations: o = 18623 (o.e = 9333, o.c = 9290)
Number of events: e = 1878

                        RR       95%-CI     z p-value
Common effect model   0.85 [0.78; 0.92] -3.75  0.0002
Random effects model  0.88 [0.77; 1.02] -1.73  0.0842
Prediction interval       [0.61; 1.29]

Quantifying heterogeneity (with 95%-CIs):
 tau^2 = 0.0162 [0.0000; 0.2158]; tau = 0.1272 [0.0000; 0.4645]
 I^2 = 54.2% [0.0%; 81.6%]; H = 1.48 [1.00; 2.33]

Test of heterogeneity:
     Q d.f. p-value
 10.91    5  0.0531
```

Meta-analysis - Forest plot

possible plotting functions `metafor::forest` or `meta::forest`

```
1  meta::forest(plot_Plato)
```



Do we understand these summary statistics and their difference?

Hierarchical models

McGill

Summary statistics

Fixed effect meta-analysis assumes a common treatment effect in each study and variation in observed study estimates is due only to chance, **within study variation**. The summary result provides the best estimate of an **assumed common treatment effect**

Random effects meta-analysis assumes the true treatment effect differs from study to study and provides an estimate of the average treatment effect considers both **within and between study variation**). The summary result gives the **average from the distribution of treatment effects across studies**

Prediction interval aids the interpretation of random effects meta-analysis by providing a predicted range for the true treatment effect in an individual (next) study

Bayesian meta-analysis

Maximum likelihood approach treats parameters as fixed
Ignores uncertainty in τ and confidence intervals for μ are too narrow
Bayesian model addresses this concern

```r
library(tidybayes);library(brms)
# non-informative default prior
brm_out <- brm(lrr | se(lse) ~ 1 + (1|Region), #<<
               data = dat, iter = 5000, warmup = 2000, cores = 4, refresh=0)   # #prior = set_prior("normal(0,10)", class = "sc
post <- brm_out %>%
  spread_draws(b_Intercept, r_Region[Region,]) %>%
  median_qi(condition_mean = b_Intercept + r_Region, .width = c(.95)) %>%
  mutate(Region=factor(Region, levels=c("Other","Latin.America","Asia", "Western.Europe", "Eastern.Europe","US" )))
cbind(post[,1], exp(post[,c(2:4)]))
```

```r
cbind(post[,1], exp(post[,c(2:4)]))
```

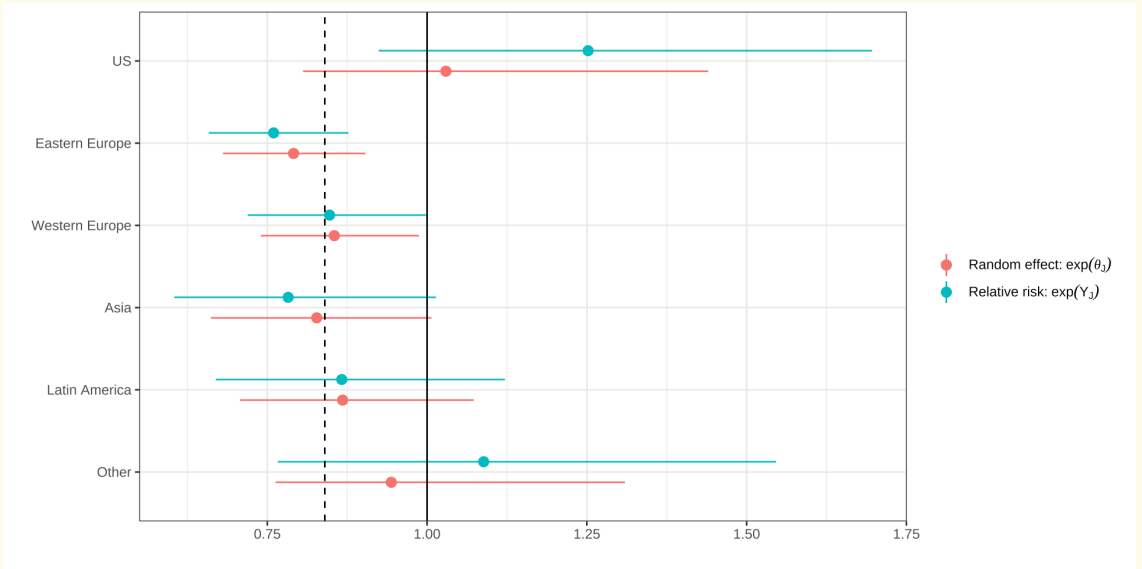
```
          Region condition_mean    .lower    .upper
1           Asia      0.8273285 0.6617992 1.0068940
2 Eastern.Europe      0.7910961 0.6807092 0.9035391
3  Latin.America      0.8678946 0.7072234 1.0732217
4          Other      0.9440098 0.7629654 1.3095628
5             US      1.0294626 0.8059452 1.4398276
6 Western.Europe      0.8548879 0.7400582 0.9872230
```

Notice shrinkage of individual region effects towards overall mean

Bayesian meta-analysis

Data Visualization of shrinkage -> global mean (dotted line) with improved variance 2^o **borrowing** of information

```
p.dat <- data.frame(lower = post$.lower, rr = post$condition_mean, upper = post$.upper)
p.dat <- exp(p.dat)
p.dat$Region <- factor(c("Asia","Eastern Europe", "Latin America", "Other", "US","Western Europe"))
p.dat <- rbind(p.dat, dat[, c("lower", "rr", "upper", "Region")])
p.dat$lab <- rep(c("Theta", "Y"), each = 6)
p.dat$id <- c(4,2,5,6,1,3,1,2,3,4,5,6)
ggplot(p.dat, aes(x = factor(Region,levels=c("Other", "Latin America", "Asia", "Western Europe", "Eastern Europe", "US")), y =
    geom_pointrange(aes(col = lab), position = position_dodge(width = 0.50)) +
    coord_flip() + geom_hline(aes(yintercept = 0.84), lty = 2) +  xlab("") +
    ylab("")  + theme(legend.position="bottom") + geom_hline(aes(yintercept = 1), lty = 1) +
    scale_colour_discrete(name="",
      labels = c("Theta" = bquote("Random effect:"~exp(theta[J])~" "),
       "Y"= bquote("Relative risk:"~exp(Y[J])))) +
  theme_bw()
```



Bayesian meta-analysis - Predictions

Easy in a Bayesian framework since study effects are assumed to be exchangeable

Simulate the posterior of $\tilde{\theta}_j$ by drawing $\tilde{\theta_j} \sim N(\mu, \tau)$ using μ and τ from the posterior

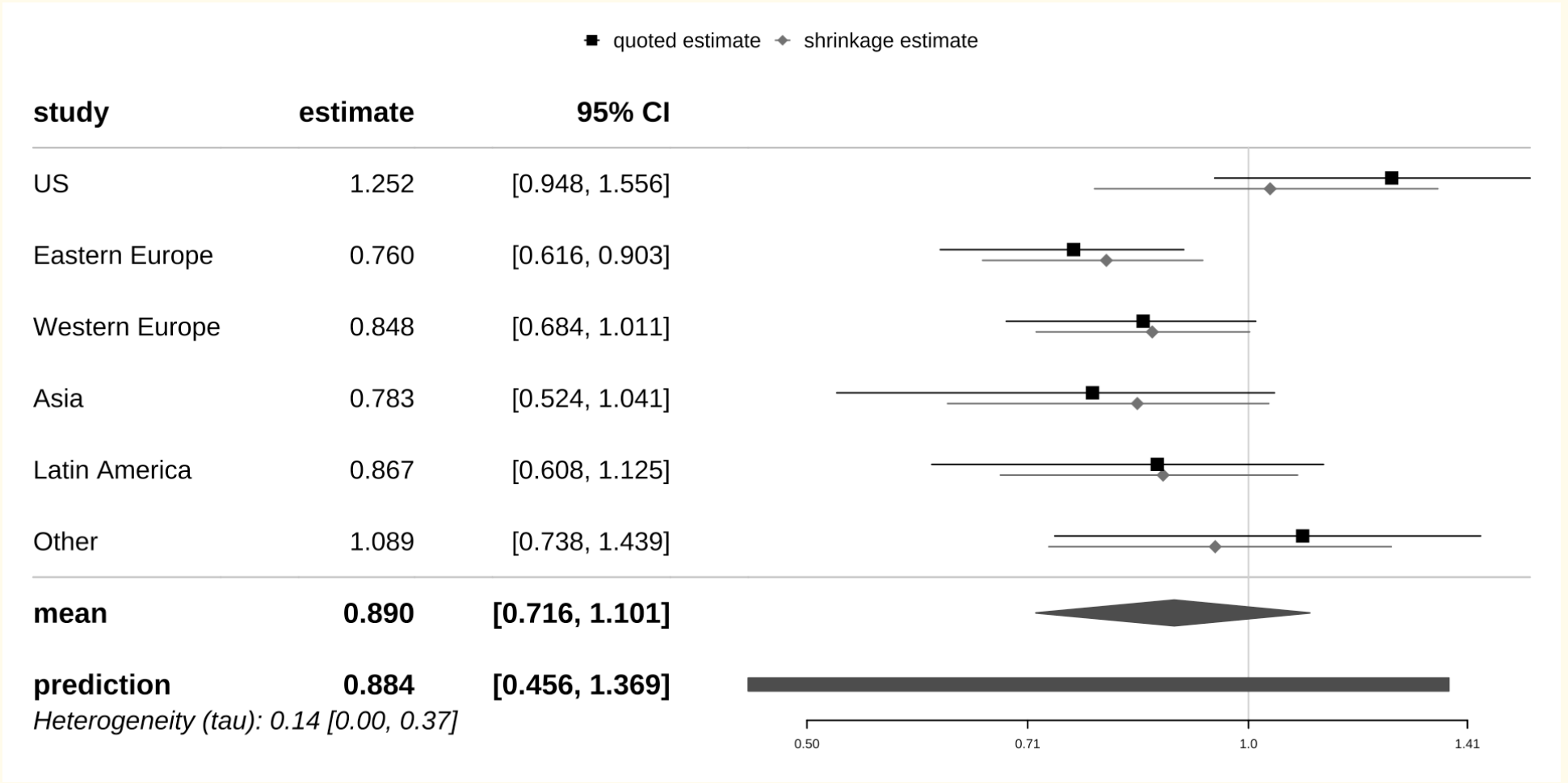
```r
library(bayesmeta)
ma01 <- bayesmeta(y = exp(dat[,"lrr"]), sigma = dat[,"lse"], labels = dat[,"Region"], mu.prior.mean = 0, mu.prior.sd = 4, tau.p
ma01$theta; ma01$summary # summary
```

```
                US Eastern Europe Western Europe      Asia Latin America
y         1.2519580     0.75987655     0.84751053 0.7827260    0.86654589
sigma     0.1549170     0.07320748     0.08359779 0.1320247    0.13178799
mode      1.0003205     0.80668934     0.86023523 0.8502698    0.86891932
median    1.0344567     0.79999914     0.85972569 0.8398095    0.87445741
mean      1.0480330     0.79747707     0.85937876 0.8346058    0.87624220
sd        0.1530569     0.06993446     0.07175383 0.1015066    0.09956671
95% lower 0.7853382     0.65875763     0.71652872 0.6235269    0.67742674
95% upper 1.3458635     0.93052361     1.00151553 1.0321938    1.07999519
              Other
y         1.0885769
sigma     0.1789770
mode      0.8952275
median    0.9490039
mean      0.9666059
sd        0.1346969
95% lower 0.7308986
95% upper 1.2510222

               tau         mu     theta
mode      0.1151484 0.87873048 0.8669743
median    0.1411086 0.88993912 0.8843447
mean      0.1610604 0.89668778 0.8966878
sd        0.1128046 0.09628676 0.2190497
95% lower 0.0000000 0.71619581 0.4560304
95% upper 0.3708797 1.10094337 1.3694340
```

Bayesian meta-analysis - Predictions

Predictions can also be displayed graphically

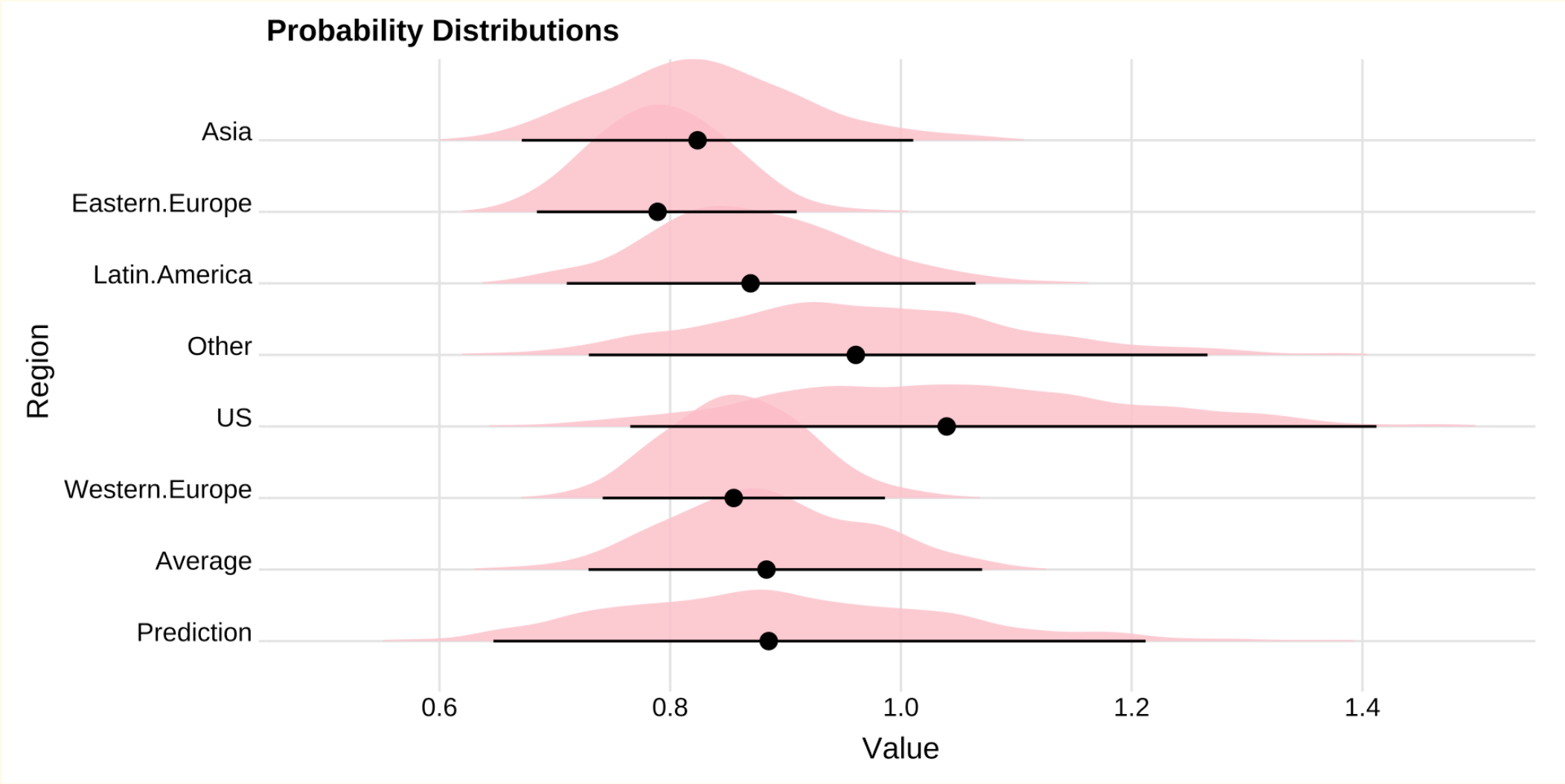
```
1 forestplot(ma01, xlog=TRUE) #adds vertical line at x=1
```



Compare to MLE - random effects 0.88 (0.77 - 1.02), prediction interval 0.88 (0.59 - 1.33)

Bayesian meta-analysis - Predictions

Predictions can also be displayed graphically using `ggridges` package



Bayesian meta-analysis

Standard Bayesian meta-analysis model follows a hierarchical approach

$\beta_j = \mu + \mu_j$ where error term $\mu_j \sim normal(0, \tau)$

μ is the average treatment effect in the (hypothetical) superpopulation of similar studies, β_j is the effect in the j-th individual study and the population variance τ^2 is referred to as the heterogeneity and

$b_j = \beta_j + \epsilon$ where error term $\epsilon_j \sim normal(0, s_j)$

models the uncertainty of the estimates from each individual study, assuming the estimate b_j from the j-th study is unbiased and normally distributed with standard error s_j.

Bayesian meta-analysis

While the observed treatment effect of a single unbiased clinical trial estimates the underlying "true" treatment effect, it is incorrect to view this as an immutable property of the treatment

Many reasons exist to expect additional variation among the underlying effects from differences between study populations, application of the treatment and measurement protocols among other factors if the study was repeated.

Leads to a paradox whereby the precision for a single trial (with less data) is better than for a meta-analysis of several trials (with more data) due to the consideration of the between study heterogeneity, τ, in the meta-analysis

Bayesian meta-analysis of a single trial!

To address this paradox Gelman has proposed using the distribution of treatment effects and heterogeneity among the 1,636 meta-analyses from the Cochrane Database of Systematic Reviews (CDSR) as prior information.

Using simulated data and cross validation, they show that their Bayesian "meta-analyses of single studies" perform much better than naively considering the single trial, which equates to setting $\tau = 0$.

The CDSR prior on the heterogeneity results in better quantification of the uncertainty, reducing the mean squared error both for estimating the study-level and population-level effects by about 60% on average, the equivalent to more than doubling the sample size!

Hierarchical modeling can aid in causal inference

Causal inference can be formulated statistically as a missing-data problem & hierarchical modeling aids in causal inference by

1. **Accounting for data collection:** In any data analysis, it is appropriate to account for any individual or group characteristics that are predictive of treatment assignment and inclusion in the dataset
2. **Adjusting for unmeasured covariates:** In structured data, multilevel modeling can yield more efficient estimates than classical no-pooling estimates
3. **Modeling variation in treatment effects:** Can model not only the expected treatment effect as a function of pre-treatment covariates x, but can also model the unexplained variance in the treatment effect

References

1. Bayes' Rules - Chapter 16 (Normal) Hierarchical Models
2. Bayesian Meta-Analysis with R and Stan

Session info

```
1 sessionInfo()
```

```
R version 4.5.2 (2025-10-31)
Platform: aarch64-apple-darwin20
Running under: macOS Sequoia 15.6.1

Matrix products: default
BLAS:   /System/Library/Frameworks/Accelerate.framework/Versions/A/Frameworks/vecLib.framework/Versions/A/libBLAS.dylib
LAPACK: /Library/Frameworks/R.framework/Versions/4.5-arm64/Resources/lib/libRlapack.dylib;  LAPACK version 3.12.1

locale:
[1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8

time zone: America/Toronto
tzcode source: internal

attached base packages:
[1] grid      stats     graphics  grDevices utils     datasets  methods
[8] base

other attached packages:
 [1] ggridges_0.5.6       epiR_2.0.85          survival_3.8-3
 [4] ggthemes_5.1.0       loo_2.8.0            baggr_0.7.11
 [7] brms_2.22.0          modelsummary_2.4.0   tidybayes_3.0.7
[10] bayesmeta_3.4        mvtnorm_1.3-3        metafor_4.8-0
[13] numDeriv_2016.8-1.1  metadat_1.4-0        forestplot_3.1.7
```