

# Bayesian Hierarchical Modeling in Stan

Dina Sinclair

21 June 2018

# Outline

## Introduction

## Maths

Frequentism

Bayesianism

Hierarchical Models

## Computation

Monte Carlo Methods

Markov Chains

Stan

## Applications

City Model

Takeaways

Bayesian?? Hierarchical Modeling?? Stan??



# Frequentist Approach

Frequentists base their conclusions on *significance testing*. This involves:

- ▶ A hypothesis to test
- ▶ A significance level  $\alpha$
- ▶ a p-value
- ▶ statistical significance when  $p < \alpha$

# Bayesian Approach

Bayesians like to use *Bayes' theorem*, which looks at the probability event  $\theta$  will happen given event  $y$  happened already:

$$p(\theta|y) = \frac{p(\theta)p(y|\theta)}{p(y)}$$

This allows us to condition on  $y$ , our prior knowledge.

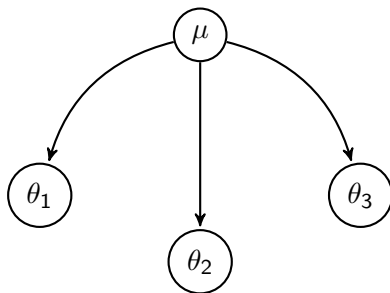


*“To the non-Bayesian, a Bayesian is someone who pollutes clean data with a subjective prior distribution. But, to the Bayesian, a classical statistician is someone who arbitrarily partitions all the available information into something called “the data” which can be analyzed and something called “prior information” which is off limits” - Andrew Gelman*



## Hierarchical Models

Bayesian models can also impose *structure* on the data.



# Computational Problems Begin

So what happens if you're calculating your Bayesian probabilities and you get stuck trying to find  $p(y)$ , where

$$p(y) = \int_{\theta} p(y|\theta)p(\theta)d\theta$$

The problem: Calculating this integral can get messy! <sup>1</sup>

---

<sup>1</sup>Dustin Stansbury, *The Clever Machine*



# Monte Carlo Methods for Integration

Expected value of some function  $f(x)$  of a random variable  $X$  distributed through  $p(x)$ :

$$E[x] = \int_{p(x)} p(x)f(x)dx \approx \int_{\theta} p(y|\theta)p(\theta)d\theta$$

Which has the same structure as our first integral.

# Monte Carlo Methods for Integration

Since we can approximate expected values using

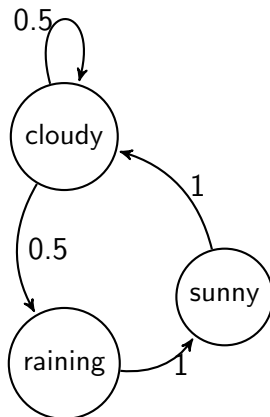
$$E_{p(x)} [f(x)] \approx \frac{1}{N} \sum_{i=1}^N f(x_i)$$

we can do the same here!

$$E_{\theta} [p(y|\theta_i)] \approx \frac{1}{N} \sum_{i=1}^N p(y|\theta_i)$$

# Markov Chain Sampling

So how do we sample from a distribution quickly and effectively?  
Markov chains!



So coding this sounds... fun, right?

So coding this sounds... fun, right?

That depends on the *language* you code in.

# Java

```
public class HelloWorld
{
    public static void main (String[] args)
    {
        System.out.println("Hello, world!");
    }
}
```



## Java

```
public class HelloWorld
{
    public static void main (String[] args)
    {
        System.out.println("Hello, world!");
    }
}
```

## Python

```
print "Hello, world!"
```



```

global hfcs= "Endline_HFC_Output"

*global field_data = "appended_forDina_deid.dta"
global field_data = "ASP Endline_combined_labeled_NO_PII.d
> ta"

set maxvar 20000

cd "~/Box Sync/3 Deidentified/"
C:\Users\Dina\Box Sync\3 Deidentified

use "ASP Endline_combined_labeled_NO_PII.dta", replace

cd "~/Box Sync/2 Data/"
C:\Users\Dina\Box Sync\2 Data

.
.
*****
***** TO CHANGE WHEN RUNNING AGAIN *****
*****
.
global today "19062018check"

end of do-file

Command

```

```

program define get_yesno
  args varname i question
  count if missing(varname)
  scalar varname=100*r(N)/N
  count if !missing(varname)
  scalar varname=v = r(N)
  count if varname == 1
  scalar varname'y = 100*r(N)/varname*v
  count if varname == 0
  scalar varname'n = 100*r(N)/varname*v
  local ++i
  putexcel A'i==" B'i=" 'question' (for varname)'
  local ++i
  putexcel A'i==" % Missing B'i="varname'm, nformat(number_d2)
  local ++i
  putexcel A'i==" % Yes !NA B'i="varname'y, nformat(number_d2)
  local ++i
  putexcel A'i==" % No !NA B'i="varname'n, nformat(number_d2)
end

program define get_dkref
  args varname i question
  count if missing(varname)
  scalar varname'nm = r(N)
  count if varname == -99
  scalar varname'dk=100*r(N)/varname'nm
  count if (varname == -98)
  scalar varname'rf=100*r(N)/varname'nm
  local ++i
  putexcel A'i==" B'i=" 'question' (for varname)', nformat(number_d2)
  local ++i
  putexcel A'i==" % Ref B'i="varname'rf, nformat(number_d2)
  local ++i
  putexcel A'i==" % DK B'i="varname'dk, nformat(number_d2)
end

```



The screenshot shows the RStudio interface with a script editor containing Stan code and a console window showing the output of the model.

**Script Editor:**

```

450 theta_updated_Single <- posteriors_Single[str_detect(colnames(posteriors_Single), 'theta')] # Vector of posterior theta values
451 theta_star_updated_Single <- sort(theta_updated_Single)[1:n_implements] # Determine n_implements lowest values
452 city_star_updated_Single <- paste(sort(sapply(theta_star_updated_Single, FUN = function(x) which(theta_updated_Single == x))), collapse=', ') # Determine cities corresponding
453
454 informationValueDFUpdatedCity_Single[jjj] <- city_star_updated_Single
455 informationValueDFCitiesDiff_Single[jjj] <- informationValueDFUpdatedCity_Single[jjj] != informationValueDFPriorCity[jjj] # Correct form of equality?
456 informationValueDFUpdatedBenefit_Single[jjj] <- sum(posteriors_Single[, paste0('theta', unlist(str_split(informationValueDFUpdatedCity_Single[jjj], pattern = ", "))))
457
458 # Update theta_hat of prior city*
459 informationValueDFPriorBenefitPrime_Single[jjj] <- sum(posteriors_Single[, paste0('theta', unlist(str_split(informationValueDFPriorCity[jjj], pattern = ", "))))
460
461 # What is change in theta^* value?
462 # Always nonpositive b/c if prior selection of cities is still best, difference is zero, if it's worse, than the new selection must improve benefit
463 # Or nonnegative depending on sign of benefit
464 informationValueDFBenefitDiff_Single[jjj] <- informationValueDFUpdatedBenefit_Single[jjj] - informationValueDFPriorBenefitPrime_Single[jjj]
465
466 # Store tau2 value
467 informationValueDFtau2_overall[jjj] <- posteriors_Single[, 'tau2']
468
469 # Rank pilot group
470 pilotRanks_Single <- match(pilotLocations, order(theta_updated_Single))
471 informationValueDFPilotGroupRanking_Single[jjj] <- sum(pilotRanks_Single)
472 informationValueDFSumPilotInTop5_Single[jjj] <- sum(pilotRanks_Single <= 5)
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

**Console:**

```

~/Oxford/Evaluating Prior Impact/ iD
tau      1.36    0.03 0.53 0.76 1.01 1.24 1.33 2.79 317 1.01
theta[1] 7.89    0.00 0.11 7.70 7.82 7.89 7.97 8.11 762 1.00
theta[2] 10.04   0.01 0.39 9.22 9.77 10.02 10.30 10.75 696 1.00
theta[3] 9.49    0.01 0.20 9.11 9.35 9.49 9.63 9.87 754 1.00
lp_____ -4.79    0.09 1.66 -8.67 -5.82 -4.31 -3.53 -2.44 334 1.00

Samples were drawn using MCMC(diag_n) at Sun Jun 17 18:35:57 2018.
For each parameter, n_eff is a crude measure of effective sample size,
and Rhat is the potential scale reduction factor on split chains (at
convergence, Rhat=1).

```



File Edit Selection Find View Goto Tools Project Preferences Help

```
randomEffectsModelConstrained.stan
1 data {
2   int<lower=0> J; // number of individuals per study
3   int<lower=0> I; // number of studies
4   real Y[I, J]; // data points from study i individual j
5   real<lower=0> sigmaSq[I, J]; // var of effect estimates
6 }
7 parameters {
8   real mu;
9   real<lower=0> tau;
10  real theta[I];
11 }
12 transformed parameters {
13 }
14 model {
15   for (i in 1:I){
16     target += normal_lpdf(theta[i] | mu, tau^2);
17     for (j in 1:J){
18       target += normal_lpdf(Y[i, j] | theta[i], sigmaSq[i, j]);
19     }
20   }
21 }
22
```

Line 9, Column 21

Spaces: 2

Plain Text

## Why choose Stan?

- ▶ Free, open source
- ▶ Faster to write
- ▶ Faster to run
- ▶ Easier to learn
- ▶ Easier to understand/reproduce other people's code

We want to implement a program in a subset of cities. Where will be the most effective to implement it?



We want to implement a program in a subset of cities. Where will be the most effective to implement it?

Initial priors



We want to implement a program in a subset of cities. Where will be the most effective to implement it?

Initial priors → Pilot programs



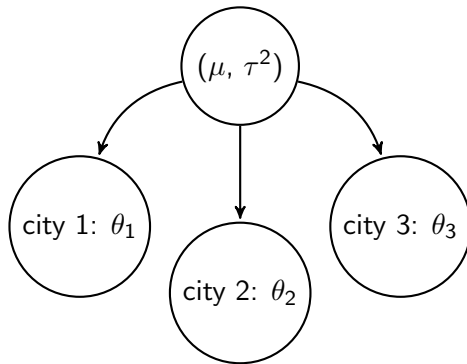
We want to implement a program in a subset of cities. Where will be the most effective to implement it?

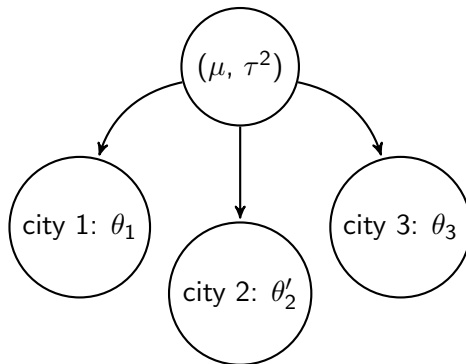
Initial priors  $\rightarrow$  Pilot programs  $\rightarrow$  Updated priors

We want to implement a program in a subset of cities. Where will be the most effective to implement it?

Initial priors → Pilot programs → Updated priors → Action







The city example is simple, but it gets at some key ideas.

- ▶ Hierarchical modeling can help us improve experimental design.
- ▶ With Stan, hierarchical modeling is a realistic path.
- ▶ Statistics and other fields use these tools regularly - why not economics?

For those that are curious, Stan has a great website and growing number of tutorials at [mc-stan.org](http://mc-stan.org)

If all goes well, my cities example will be online by the time I've left GPI as well!