



Using data to detect environmental dangers

California's pesticide hotspots

Michael Corey, CIR @mikejcorey



DANGER PELIGRO
Pesticidas Pesticidas
Area Under Fumigation
Methyl Bromide and Chloropicrin
Fumigant in USE
DO NOT ENTER
NO ENTRE

DANGER PELIGRO
Pesticides Pesticidas
Area Under Fumigation
Methyl Bromide and Chloropicrin
Fumigant in USE
DO NOT ENTER
NO ENTRE

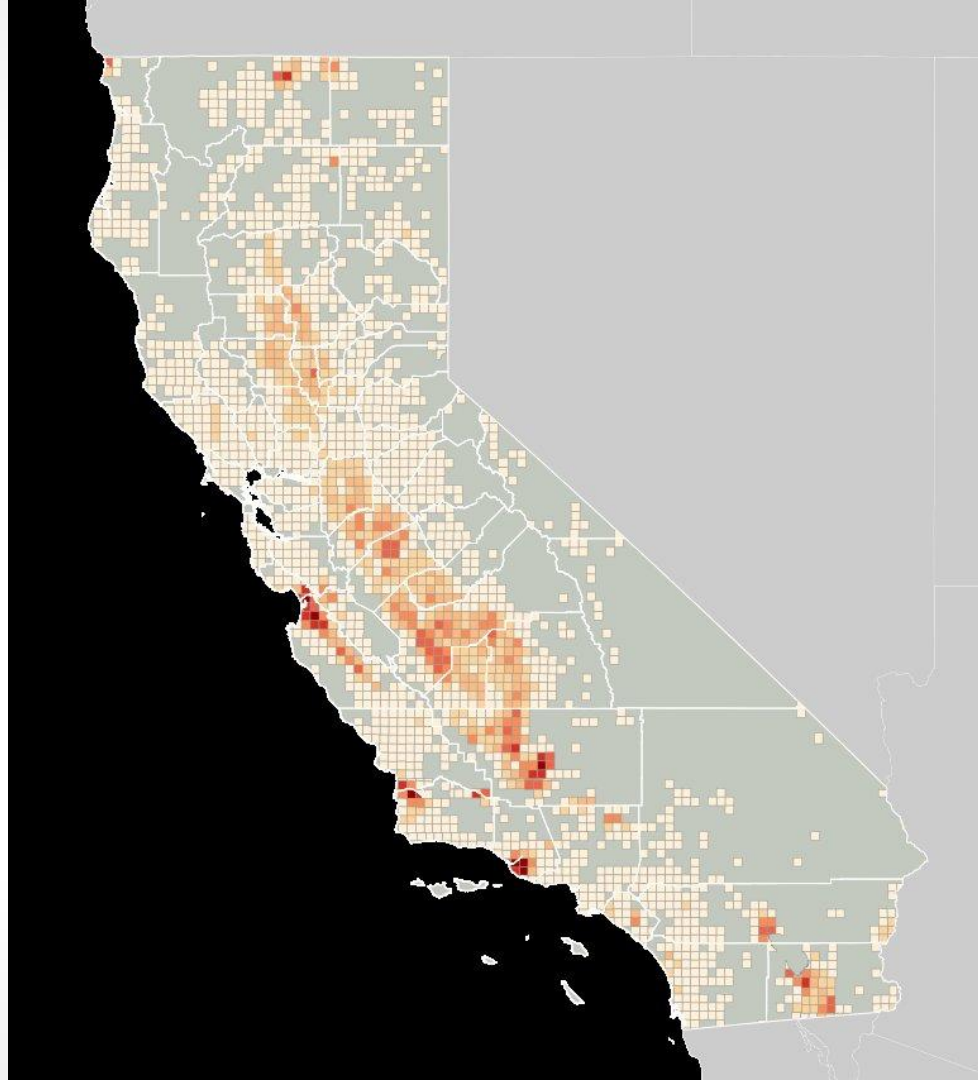










































10 years of pesticides

- 22 million individual applications
- 1.5 billion pounds of pesticides



Get the right data

- Found out (after we wrote a lot of code) that the public data on their website was outdated and had uncorrected errors

Name	Size	Date Modified
 [parent directory]		
 README.txt	172 B	3/14/12, 5:00:00 PM
 pur1974.zip	10.3 MB	7/22/09, 5:00:00 PM
 pur1975.zip	10.9 MB	7/22/09, 5:00:00 PM
 pur1976.zip	10.6 MB	7/22/09, 5:00:00 PM
 pur1977.zip	11.1 MB	7/22/09, 5:00:00 PM
 pur1978.zip	8.7 MB	7/22/09, 5:00:00 PM
 pur1979.zip	12.4 MB	7/22/09, 5:00:00 PM
 pur1980.zip	11.2 MB	7/22/09, 5:00:00 PM
 pur1981.zip	13.7 MB	7/22/09, 5:00:00 PM
 pur1982.zip	12.1 MB	7/22/09, 5:00:00 PM
 pur1983.zip	277 MB	10/24/14, 1:29:00 PM
 pur1984.zip	13.4 MB	7/22/09, 5:00:00 PM
 pur1985.zip	16.9 MB	7/22/09, 5:00:00 PM
 pur1986.zip	19.5 MB	7/22/09, 5:00:00 PM
 pur1987.zip	20.5 MB	7/22/09, 5:00:00 PM
 pur1988.zip	18.1 MB	7/22/09, 5:00:00 PM
 pur1989.zip	18.9 MB	7/22/09, 5:00:00 PM
 pur1990.zip	37.7 MB	7/22/09, 5:00:00 PM
 pur1991.zip	68.9 MB	7/22/09, 5:00:00 PM
 pur1992.zip	76.9 MB	7/22/09, 5:00:00 PM
 pur1993.zip	81.0 MB	7/22/09, 5:00:00 PM
 pur1994.zip	91.4 MB	7/22/09, 5:00:00 PM
 pur1995.zip	62.8 MB	7/22/09, 5:00:00 PM
 pur1996.zip	54.9 MB	7/22/09, 5:00:00 PM
 pur1997.zip	56.6 MB	7/22/09, 5:00:00 PM
 pur1998.zip	56.3 MB	7/22/09, 5:00:00 PM
 pur1999.zip	54.2 MB	7/22/09, 5:00:00 PM
 pur2000.zip	55.2 MB	7/22/09, 5:00:00 PM
 pur2001.zip	64.2 MB	7/22/09, 5:00:00 PM
 pur2002.zip	87.6 MB	6/23/14, 5:00:00 PM
 pur2003.zip	49.7 MB	6/23/14, 5:00:00 PM
 pur2004.zip	50.6 MB	7/22/09, 5:00:00 PM
 pur2005.zip	55.3 MB	7/22/09, 5:00:00 PM
 pur2006.zip	57.0 MB	7/22/09, 5:00:00 PM
 pur2007.zip	57.1 MB	4/19/09, 5:00:00 PM
pur2008.zip	96.0 MB	1/6/10, 4:00:00 PM
pur2009.zip	101 MB	12/27/10, 4:00:00 PM
pur2010.zip	108 MB	1/16/12, 4:00:00 PM
pur2011.zip	130 MB	10/24/14, 1:29:00 PM
pur2012.zip	134 MB	10/24/14, 1:30:00 PM

Record Structures for UDC and Lookup Table data

The following tables define the record structure of each data file. The "Field Seq. No." (field sequence number) identifies the order in which each field appears in the data record and in the data dictionary in Chapters 2 and 3 of this document. The "field name" indicates the name of the field. The "type" indicates whether the field is a Numeric (N), Character (C), or Date (Date) field. "Mask" displays the field as Numeric, Character, or Date values representing the size of each field along with decimal places (if used).

Use Data Chemical (UDC)

Field Seq. No.	Field Name	Type	Mask
1	USE_NO	N	N(8)
2	PRODNO	N	9999999
3	CHEM_CODE	N	99999
4	PRODCHEM_PCT	N	999.99999
5	LBS_CHM_USED	N	Floating Decimal
6	LBS_PRD_USED	N	N(10).9999
7	AMT_PRD_USED	N	N(8).9999
8	UNIT_OF_MEAS	C	AA
9	ACRE_PLANTED	N	N(8).99
10	UNIT_PLANTED	C	A
11	ACRE_TREATED	N	N(8).99
12	UNIT_TREATED	C	A
13	APPLIC_CNT	N	9999999
14	APPLIC_DT	DATE	MMDYYYY
15	APPLIC_TIME	N	HHMM
16	COUNTY_CD	C	AA
17	BASE_LN_MER	C	A
18	TOWNSHIP	C	AA
19	TSHIP_DIR	C	A
20	RANGE	C	AA
21	RANGE_DIR	C	A
22	SECTION	C	AA
23	SITE_LOC_ID	C	A(8)

What does it mean?

- RTFM
- Talk to the people who made it
- See what others have done with it

- pesticides
 - apps
 - chemicals
 - management
 - commands
 - __init__.py
 - cache_api_jsons.py
 - check_chems_of_concern.py
 - clear_errors.py
 - date_series.py
 - export_rejected_apps.py
 - generate_concern_cats.py
 - load_crops_only.py
 - load_pesticides.py
 - test_counties.py
 - update_ag_uses.py
 - update_app_errors.py
 - update_fumigant_applications.py
 - update_fumigants_used.py
 - update_stats.py
 - __init__.py
 - __init__.py
 - admin.py
 - models.py
 - tests.py
 - utils.py
 - views.py
 - core
 - countymap
 - grid
 - management
 - commands
 - __init__.py
 - calculate_grid_percentiles.py
 - calculate_percentiles.py
 - check_geojson_values.py
 - compare_grid_totals.py
 - compare_jenks_years.py
 - dump_api_tables.py
 - dump_multicounty_grids.py
 - find_all_top5_chemicals.py
 - get_random_13d_grids.py
 - get_random_grids.py
 - load_13d_overages.py
 - load_county_boundaries.py

```

22
23 working_dir = os.path.join(settings.SITE_ROOT, 'data', '20140826')
24 minyear = 2003
25 maxyear = 2012
26
27 def mk_decimal(self, value):
28     value = value.strip()
29     try:
30         return Decimal(value)
31     except:
32         return None
33
34 def remove_spaces(self, value):
35     value = value.strip()
36     return value.replace(' ', '')
37
38 def load_chemicals(self, year_path):
39     logger.info('Starting to load chemicals ...\n')
40
41     file_path = os.path.join(year_path, 'chemical.txt')
42     csv_file = open(file_path, 'rU')
43     csv_object = csv.DictReader(csv_file, delimiter=',', quotechar='"')
44     for row in csv_object:
45         chem_code = row['chem code'].strip()
46         chem_name = row['chemname'].strip()
47         chemical, chemical_created = Chemical.objects.get_or_create(
48             code=chem_code,
49             name=chem_name,
50         )
51     csv_file.close()
52     logger.info('Finished loading chemicals.\n')
53
54     logger.info("Generating categories of concern ...")
55     call_command('generate_concern_cats', interactive=True)
56
57     logger.info("Marking fumigants ...")
58     for c in Chemical.objects.all():
59         if c.check_fumigant():
60             c.fumigant = True
61             c.save()
62
63 def load_products(self, year_path):
64
65     logger.info('Starting to load products ...\n') # helps you know where it is in process
66
67     # load product
68     file_path = os.path.join(year_path, 'product.txt')
69     csv_file = open(file_path, 'rU')
70     csv_object = csv.DictReader(csv_file, delimiter=',', quotechar='"')
71     for row in csv_object:
72         if row['fumigant sw'].strip().lower() == 'x': # NOTE: there appears to be at least 1 weird value of "01011901" in 2002, but ignoring for now
73             bool_fumigant = True
74         else:
75             bool_fumigant = False
76         product, product_created = Product.objects.get_or_create(
77             regno=self.remove_spaces(row['show regno']), # In some file somewhere bad spaces got put in here
78             name=row['product name'].strip(),
79             prodno=row['prodno'].strip(),
80             fumigant=bool_fumigant, # probably not very useful
81         )

```

What is important?



All applications

Agricultural use

No errors

Dangerous chemicals

Chemical Code	Name	Category						Pounds applied
		PRIOR	TAC	FUM	CARC	REP/DEV	CHOIN	ALL
00629	Ziram	*						4,507
00211	Mancozeb	*	*		*			3,627
03946	Glufosinate-ammonium	*						3,371
01973	Oxyfluorfen	*						3,091
01868	Oryzalin	*						2,690
02081	Iprodione	*			*			2,414
00531	Simazine	*						2,366
00231	Diuron	*			*			2,191
00806	2,4-D, dimethylamine salt		*					2,054
00445	Propargite	*			*	*		1,964
00198	Diazinon	*					*	1,785
00070	Bensulide	*					*	1,718
00383	Methomyl						*	1,539
01685	Acephate	*					*	1,493
00418	Naled	*	*				*	1,352
04022	Propamocarb hydrochloride						*	1,321
00216	Dimethoate	*					*	1,259
00179	Chlorthal-dimethyl	*						1,190
02008	Permethrin	*			*			1,174
00382	Oxydemeton-methyl	*				*	*	1,173
04000	Cyprodinil	*						1,124
01626	Ethephon						*	1,074
05759	Pyraclostrobin	*						1,058

Agricultural Pesticide Use Near Public Schools in California

California Environmental Health Tracking Program

April 2014



Does what you find make sense?

- Found state couldn't be rigorously checking data that Dow (!) was maintaining for them.
- Double-team with reporter to see if we independently arrive at same numbers
- Watch out: Have regulations changed over time?

What is useful for us to do?

- Show people what had been applied near them
- Show people the most-affected parts of the state
- Identify people in most-affected parts of state, contact them directly

Things we could only say because of the data

- More than 1 million people affected by overages
- How many times the state exceeded its own guidelines
- Six places exceeded every year
- One place exceeded by over 1 million pounds
- Strawberries account for 8 percent of state pesticide use (but only 1 percent of total farmland)
- Strawberry growers' use of 1,3-D tripled over 10 years