InterPoll: Crowd-Sourced Internet Polls (Done Right)

MSR-TR-2014-3

Benjamin Livshits and Todd Mytkowicz
Microsoft Research

Abstract

Crowd-sourcing is increasingly being used for providing answers to online polls and surveys. However, existing systems, while taking care of the mechanics of attracting crowd workers, poll building, and payment, provide little that would help the survey-maker or pollster to obtain statistically significant results devoid of even the obvious selection biases.

This paper proposes InterPoll, a platform for programming of crowd-sourced polls. Polls are expressed as embedded LINQ queries, whose results are provided to the developer. InterPoll supports reasoning about uncertainty, enabling t-tests, etc. on random variables obtained from the crowd. InterPoll performs query optimization, as well as bias correction and power analysis, among other features. Making InterPoll queries part of the surrounding program allows for optimizations that take advantage of the surrounding code context. The goal of InterPoll is to provide a system that can be reliably used for research into marketing, social and political science questions.

This paper highlights some of the existing challenges and how InterPoll is designed to address most of them. We outline some of the optimizations and give numerous motivating examples designed to illustrate our system design. Note that this paper is an outline of our vision — we deliberately focus on examples and motivation and leave a detailed technical treatment for future work.

1. Introduction

Online surveys have emerged as a powerful force for assessing properties of the general population, ranging from conducting marketing studies, to product development, to political polls, to customer satisfaction surveys, to medical questionnaires. Online polls are widely recognized as an affordable alternative to in-person surveys, telephone polls, or face-to-face interviews. Psychologists have argued that online surveys are far superior to the traditional approach of finding subjects which involves recruiting college students, leading to the famous quip about psychology being the study of the college sophomore[18].

Online surveys allow one to reach wider audience groups and to get people to answer questions that they may not be comfortable responding in a face-to-face setting. While online survey tools such as Instant.ly, SurveyMonkey, Qualtrics, and Google Customer Surveys take care of the mechanics of online polling and make it easy to get *started*, the results they produce often create more questions than they provide answers [20, 23, 28, 44, 54, 124].

Surveys, both online and offline, suffer from *selection biases*, as well as non-response, and coverage issues. These biases are not trivial to correct for, yet without doing so, the data obtained from surveys may be less than representative and cannot be used for reporting. InterPoll allows the developer to both *estimate* and *correct* for the biases and errors inherent in the data they are collecting.

It is also not so obvious how many people to poll. Indeed, polling too few yields results that are not statistically significant; polling too many is a waste of money. None of the current survey platforms help the survey-maker with deciding on the appropriate number of samples. Today's online survey situation can perhaps be likened to playing slot machines with today's survey sites playing the role of a casino; it is clearly in the interest of these survey sites to encourage more polls being completed.

In addition to the issue of data quality and representativeness, *cost* of the polls is an important consideration for poll makers, especially given that thousands of participants may be required. Even if answering a single question can costs cents, often getting a high level of assurance for targeted population segment involves hundreds of survey takers and significant bills. In fact, deciding on how to properly target the survey is a non-trivial task: if general audience surveys cost $.10 per question and *targeted* ones cost $.50 per question, is it better to ask five times as many questions of the general audience and then post-process the results or is it better to ask fewer questions of the targeted audience? Given that demographic targeting can often involve dozens of categories (males, 20–30, employed full-time, females, 50–60, employed part-time, females, 20–30, students, etc.) how does one properly balance the need for targeted answers and the cost of reaching these audiences?

We see these challenges as interesting optimization problems. To address some of these issues, InterPoll has an optimization engine whose goals is to determine (a sequence of) questions to ask and targeting restrictions to use. The primary goal of the optimization is to get a certain level of certainty in a developer-provided question (i.e. do men aged from 30–50 prefer *Purina Dog Chow* to *Precise Naturals Grain Free*), while minimizing the cost involved in running the poll on a large scale.

This paper proposes InterPoll, a platform for in-application scripting of crowd-sourced polls, giving developers streamlined access to crowd-sourced poll data. To easy integration into with existing programs we allow InterPoll expressions to be written as LINQ queries [78]. InterPoll performs query optimization, as well as bias correction and power analysis, among other features, to enable a system that can be reliably used for research into marketing, social and political science questions.

1.1 Motivating Examples

One of the goal of InterPoll is to make running crowd-sourced polls easy for the developer. We accomplish this by using LINQ [78], language-integrated queries. LINQ is natively supported by .NET, with Java providing similar facilities with JQL.

Example 1 (Basic filtering) A simple poll may be performed the following way:

```
1  var people = new MTurkQueryable<Person>(true, 5, 100, 2);
2  var liberalArtsPairs = from person in people
3          where person.Employment == Employment.STUDENT
4          select new {
5                  Person = person,
6                  Value = person.PoseQuestion<bool>(
7                          "Are you a liberal arts major?")
8                  };
```

The first line gets a handle to a population of users, in this case obtained from Mechanical Turk, although other back-ends are also possible. Populations on which we operate have associated demographic information; for example, note that the `where` clause on line 3 ensures that we only query (college) students. This poll will ask (college) students if they study liberal arts, producing an iterator of $\langle$`Student`, `bool`$\rangle$ pairs represented in .NET as `IEnumerable`. □

Example 2 (Counting) Given `liberalArtsPairs`, it is possible to do a subsequent operation on the result, such as printing out all pairs or using, the `Count` operation to count the liberal arts majors:

```
1  var libralArtMajorsCount =
2          (from pair in liberalArtsPairs
3          where pair.Value == true
4          select person).Count();
5  double percentage = 100.0 * libralArtMajorsCount /
6                              liberalArtsPairs.Count();
```

Lines 5 and 6 compute the percentage of liberal art majors within the previously collected population. □

Example 3 (Uncertainty) InterPoll explicitly supports computing with uncertain data, using a style of programming proposed in Bornholt *et al.* [10].

```
1  var liberalArtWomen = from person in people
2    where person.Gender == Gender.FEMALE
3    where person.Employment == Employment.STUDENT
4    select person.PoseQuestion<bool>(
5      "Are you a liberal arts major?");
6
7  var liberalArtMen = from person in people
8    where person.Gender == Gender.MALE
9    where person.Employment == Employment.STUDENT
10   select person.PoseQuestion<bool>(
11     "Are you a liberal arts major?");
12
13 var femaleVar = femaleSample.ToRandomVariable();
14 var maleVar = maleSampleList.ToRandomVariable();
15 if(femaleVar > maleVar){
16   Console.WriteLine("More female liberal arts majors.");
17 }else{
18   Console.WriteLine("More male liberal arts majors.");
19 }
```

Here, we convert the Boolean output of the posted question to a random variable (lines 13 and 14). Then we proceed to compare these on line 15. Note that the implicit > comparison on line 15 actuality compiles to a t-test on `femaleVar` and `maleVar`. □

Example 4 (Explicit t-tests) Here we explicitly perform the t-test at a specified confidence interval.

```
1  var test =
2          maleSampleList.ToRandomVariable() >
3          femaleSample.ToRandomVariable();
4
5  if (test.AtConfidence(.95)) { ... }
```

The test and the confidence interval determine the outcome of a power analysis that InterPoll will perform to decide how many (male and female) subjects to poll. □

Example 5 (Optimizations) Suppose we are conducting a marketing study of dog owners' preference for Purina Puppy Chow. Specifically, we are trying decide if married women's attitude toward this product is more positive than that of married men.

```
1  var puppyChowWomen = from person in people
2    where person.PoseQuestion<bool>("Are you a dog owner?")
3          == true
4    where person.Gender == Gender.FEMALE
5    where person.Relationship == Relationship.MARRIED
6    select person.PoseQuestion<bool>(
7      "Would you consider using Purina Puppy Chow?");
```

Similarly, for men:

```
1  var puppyChowMen = from person in people
2    where person.PoseQuestion<bool>("Are you a dog owner?"))
3          == true
4    where person.Gender == Gender.MEN
5    where person.Relationship == Relationship.MARRIED
6    select person.PoseQuestion<bool>(
7      "Would you consider using Purina Puppy Chow?");
```

To compare these two, the following comparison may be used:

```
1  if(puppyChowWomen > puppyChowMen){
2    Console.WriteLine("Women like puppy chow more");
3  }
```

In this case it is not so obvious how to sample from the population: a naïve strategy is to sample women first, then sample men. However, another strategy may be to sample everyone (who is `MARRIED`) and to separate them into two streams: one for women, the other for men. Lastly, sampling from the same population is likely to yield a disproportional number of samples in either population. For example, 64% of users of the uSamp platform are women [117] as opposed to 51%, as reported by the US 2012 Census. □

Example 6 (Language integration) One of the advantages of InterPoll is its integration into the surrounding programming environment of .NET, which allows the developer to freely mix human and computer computation.

```
1  foreach(string productName in products){
2    var women = ...;
3    var men = ...;
4    if(men > women){
5      Console.WriteLine("Women like {0} more", productName);
6    }
7  }
```

Of course, fundamentally, human and computer computation have markedly different properties (for instance, latencies are much higher for crowd-sourced tasks). To illustrate

our point, above we modified the previous example to test for the respective fondness for a collection of products using a `foreach` loop in C#. □

1.2 Challenges

The examples described in the previous section raise a number of non-trivial challenges.

Query optimization: How should these queries be executed? How can the queries be optimized to avoid unnecessary work? Should doing so take the surrounding .NET code into which the queries are embedded into account? Should they be run independently or should there be a degree of reuse (or result caching) between the execution plans for the men and women? While a great deal of work on database optimizations exist, both for regular and crowd-sourced databases, much is not directly applicable to the InterPoll setting [9, 16, 43, 83, 84, 99], in that the primary goal of InterPoll optimizations is reducing the amount of money spent on a query.

Query planning: How should we run a given query on the crowd back-end? For instance, should pre-filter crowd workers or should we do post-filtering ourselves? Which option is cheaper? Which crowd back-end should we use, if they have different pricing policies? Should the filtering (by gender and relationship status) take place as part of population filtering done by the crowd provider?

Bias correction: Given that men and women do not participate in crowd-sourcing at the same rate (on some crowd-sourcing sites, one finds about 70% women and 30% men [49, 87, 94]), how do we correct for the inherent population bias to match the more equal gender distribution consistent with the US Census (typically, 51% women, 49% men)? Similarly, studies of CrowdFlower samples show a disproportionately high number of democrats vs. republicans [30]. Mobile crowd-sourcing tends to attract a higher percentage of younger participants [37].

Ignorable sample design: An unstated assumption in much of crowd-sourcing work is that of *ignorable sample design*. Ignorable designs assume that sample elements are missing from the sample when the mechanism that creates the missing data occurs at random, often referred to as missing at random or completely missing at random [93]. An example of non-ignorable design is asking what percentage of people know how to use a keyboard: in a crowd sample that need a keyboard to fill out the survey, the answer is likely to be nearly 100%; in the population as a whole it is likely to be lower.

Power analysis: Today, the users of crowd-sourcing are forced to decide how many participants or workers to use for every task, yet there is often no solid basis for such a decision: too few workers will produce results of no statistical significance; too many will result in overpayment. How many samples (or workers) are required to achieve the desired level of statistical significance?

Crowd back-end selection: Given that different crowd back-ends may present different cost trade-offs (samples stratified by age or income may be quite costly, for example) and demographic characteristics, how do we pick an optimal crowd for running a given set of queries [81]? How do we compare query costs across the back-ends to make a globally optimal decision?

Quality control: What if the users are selecting answers at random? This is especially an issue if we ask about properties that eschew independent verification without direct contact with the workers, such as their height. A possible strategy is to insert attention-checking questions also called "catch trials" and the like [52, 86].

Privacy of human subjects: Beyond the considerations of ethics review boards (IRBs) and HIPAA rules for health-related polls, there is a persistent concern about being able to de-anonymize users based on their partial demographic and other information. A famous study by Sweeney was able to identify Governor Weld (then governor of the state of Massachusetts) via his gender, ZIP code, and birth date [111].

1.3 Contributions

Our paper makes the following contributions.

Declarative language-integrated queries: LINQ's language integration facilities allow us to integrate a crowd back-end as one of the providers of data within the program, obviating the need for domain-specific languages, advocated by others [4]. LINQ queries over crowd data are computed *lazily*, on demand, based on what computation (such as a t-test) needs to be applied to query results, enabling *code-sensitive* query optimizations, customized to how the data is actually used within the program.

Programming with uncertainty: Type `Uncertain<T>` previously introduced by Bornholt *et al.* [10] is natively supported in InterPoll, allowing the developer to reason about population samples in a statistical setting directly within the general-purpose language of their choice, such as C#. For instance, an `if` on two random variables is automatically converted in InterPoll to a t-test. While Bornholt *et al.*'s motivation was operating on sensor-sourced data, we observe that uncertainty and crowd-sourced data mesh particularly well.

Query optimizations and planning: We propose a range of query optimizations and strategies for query planning. These optimizations are quite different in spirit from either traditional database query optimizations or crowd optimizations, and are enabled by the following three features of InterPoll (1) seeing the query in the context of surrounding code; (2) support for explicit cost model for crowd-based tasks; (3) explicit support for uncertainty.

Bias correction and power analysis: having access to both the queries and how they are used in the program (i.e., within a conditional) allows us to perform bias correction (or *unbiasing*) and power analysis. The goal of the former is to correct query results so as to make them representative of the population as a whole (as captured by the US Census, for example), rather than the captured sample. The purpose of power analysis is to determine the requisite number of samples (workers) to collect.

Automatic back-end selection: As part of query planning, we can also pick the appropriate back-end for a query or a set of queries. This effectively makes InterPoll into a *cross-compiler*, able to choose the best platform for a query to execute. Additionally, InterPoll can enable better *caching*, it can help to learn priors for various events of interest, etc.

Privacy analysis: Access to the entire query and its questions allows us to reason about privacy properties of the query, potentially akin to PINQ [80].

1.4 Domains

We envision InterPoll being useful in a wide variety of domains.

- **Social sciences**: social sciences typically rely on data obtained via studies and producing such data is often difficult, time-consuming, and costly [28]. Indeed, in-person interviews require recruiting the necessary number of subjects, a time-consuming process that is almost guaranteed to introduce sample biases, given that the subjects need to be located close to the interviewer, which is why a large share of studies in psychology, for instance, use college students as subjects [18]. While not a panacea, online polls provide a number of distinct advantages [59].
- **Political polls**: these are costly and require a fairly large sample size to be considered reliable. By their very nature, subjects from different geographic locales are often needed, which means that either interviewers need to cast a wide net (exit polling in a large number of districts) [8, 104] or they need to conduct a broad remote survey (such as telephone surveys) [34, 106, 125].
- **Marketing polls**: While much has been written about the upsides and downsides of online surveys [2, 24, 28, 93], the ability to get results cheaply, combined with the ease of targeting different population segments (i.e., married, high income, dog owners) makes the web a fertile ground for marketing polls. Indeed, these are among the primary uses of sites such as Instant.ly [117], Survey Monkey [47], and Google Customer Surveys [79].
- **Health surveys**: A lot of researchers have explored the use of online surveys for collecting health data [5, 29, 92, 109, 110]. While for many domains online polling is sufficient, for the health domain additional data may often be required. For this purpose, we can build *specialized crowds* that use biometric instrumentation *in addition* to online polls. For example, we can have a crowd of workers who all use activity tracking features of recent Samsung smart-phones, FitBit, or Nintendo Fit Meter some of which can be instrumented with libraries on `http://www.openyou.org/` to record a graph of their daily activities, which can be correlated with their other reported data [50]. In particular, biometric technologies embedded into smart-phones by default could help push the reach of biometric techniques beyond the so-called "quantified self" movement. We could, for example, discover if habitual coffee drinkers are more active in the afternoon.

In all of the cases above, in addition to population biases, so-called *mode effects*, i.e. differences in results caused by asking questions online vs. on the telephone vs. in person are possible [6, 13, 25, 34, 37, 53, 96, 98, 103, 106, 125, 127].

2. Overview

When it comes to InterPoll queries, two closely related areas that need to be discussed are query optimizations and query planning. Both are established areas in database literature, to a point where entire books are written about specific query planners and tuning user interactions with them [35, 85]. Recently, several projects have focuses on optimizing LINQ queries as well [9, 16, 43, 83, 84, 99]. Several projects attempt to bring the ideas of query-based processing to crowd-sourced tasks [33, 74–76, 82].

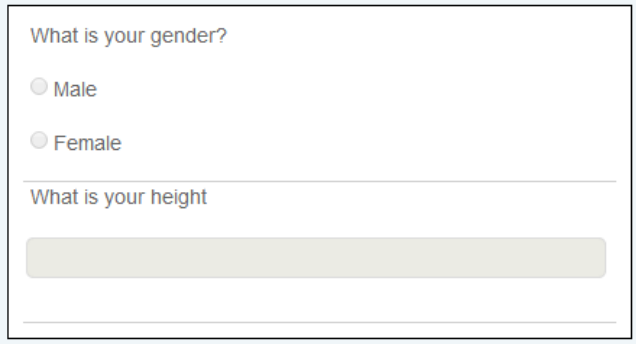


Figure 1. Sample form produced by InterPoll for the Mechanical Turk back-end.

Overall goal: The chief goal of our optimizations is the overall amount of money that needs to be spent on answering a query or a set of queries. This amount is influenced by how we optimize the query (Section 2.1); how we select a query plan (Section 2.1); how we unbias the query (Section 2.3); and by the outcome of power analysis, which decides the requisite number of samples (Section 2.4). Lastly, the choice of the back-end has a significant impact on the cost of query processing as well (Section 2.5).

2.1 Optimizations

InterPoll explicitly operates on LINQ queries that are embedded into the user program. This presents a number of powerful optimizations opportunities, reminiscent of traditional compiler optimizations, some of which we outline below.

Dead code elimination: Perhaps the simplest optimization is similar to dead code elimination in compiler literature and involves only profiling users for the demographic aspects that are needed by the query in question. For instance, the only necessary demographic characteristic that we need to obtain is the worker's employment status, because of the `where` clause

```
where person.Employment == Employment.STUDENT
```

Recursively traversing the query allows us to determine which demographic characteristics are of interest, allowing us to compile a form that needs to be shown to the worker on the Mechanical Turk back-end, as shown in Figure 1.

Constant propagation (partial evaluation): Given that LINQ queries in InterPoll are frequently used for hypothesis testing, the nature of the test can influence the test to run. For example, the conditional on line 5 below is clearly infeasible

```
1  var population50Plus =
2          from person in people
3          where person.Age > 50
4          select person.Age;
5  if (population50Plus < 20) { ... }
```

This is because we are only selecting workers 50 years or older, so there is no way the test on line 5, which involves the expected value of `population50Plus` can be successful.

Combining conditions: Conditions of nested LINQ queries can be flattened as shown below:

```
var womenOver50 =
        from person in
                (from person in population50Plus
                where person.Age > 50
                select person)
        where person.Gender == Gender.FEMALE
        select person;
```

can be rewritten as

```
var womenOver50 =
        from person in population50Plus
        where
                person.Age > 50 &&
                person.Gender == Gender.FEMALE;
        select person
```

Common sub-expression elimination: Consider the following example that involves shared sub-queries:

```
var womenOver50 =
        from person in
                (from person in people
                where person.Age > 50
                select person)
        where person.Gender == Gender.FEMALE
        select person;
var menOver50 =
        from person in
                (from person in people
                where person.Age > 50
                select person)
        where person.Gender == Gender.MALE
        select person;
var peopleOver50 = womenOver50.Concat(menOver50);
```

Here the "over 50" sub-query is shared by both the female and the male population. A more effective way to represent the same population is by eliminating gender selection

```
from person in people where person.Age > 50 select person;
```

Note that we now may end up with a different mix of women and men in our sample, which requires bias correction. The effect of this optimizations would be amplified if we were to re-evaluate the sample population within a loop.

2.2 Query Planning

Unlike traditional query planning in databases, the structure of the cost for INTERPOLL is quite different. In particular, much of the cost is actually the financial cost of hiring the requisite number of workers and having them successfully complete the polls (the two problems are in fact different, especially for long polls with dozens of questions).

Fundamentally, given a cost model, we need to select the best way to execute a given query, by running a portion of it the crowd and portion of it locally.

Example 7 (Cost analysis) Given the query below,

```
from p in people
        where p.Age > 50
        where p.Gender = Gender.FEMALE
        select p.Age;
```

the following plans are possible
To represent each plan, we use π for projection, σ for selection, and c for *crowd selection*, i.e. a form of filtering performed by the crowd itself, often at a higher cost. The table above shows each plan.

Plan	Cost	Probability
$\pi_{Age} \leftarrow \sigma_{Age} \leftarrow \sigma_{Gender}$	\$0.10	Pr[Age>50] × Pr[FEMALE]
$\pi_{Age} \leftarrow \sigma_{Age} \leftarrow c_{Gender}$	\$0.50	Pr[FEMALE]
$\pi_{Age} \leftarrow c_{Age} \leftarrow \sigma_{Gender}$	\$0.50	Pr[Age>50]
$\pi_{Age} \leftarrow c_{Age} \leftarrow c_{Gender}$	\$1	1

The second column indicates the cost of obtaining a person matching these criteria from the crowd: in our example, the cost of obtaining a person over 50 years of age from the crowd is \$0.50. The cost of obtaining a person who is over 50 *and* is female is \$1. The last column indicates the probability of obtaining such a person in the crowd population. From prior knowledge, we may know that the percentage of women in the crowd is 70%, thus making Pr[FEMALE] = 0.7. □

It is generally the case that for effective planning, one needs to know priors for a variety of joint distributions (i.e., *what is the probability of finding a married male under 20?*). If the number of σ operations that the original query compiles to is small, we can exhaustively consider all possible plans, with the expected cost of each plan obtained by multiplying the cost of an individual by the probability of obtaining them. An interesting separate problem is to try to predict the success rate of free-form questions; for instance McDonald *et al.* [79] report a success rate of 9.25% for their yes/no screening questions.

2.3 Bias Correction

It is broadly acknowledged that while crowd-sourcing platforms present a number of exciting new benefits, conclusions that may result from crowd-sourced experiments need to be treated with care [6, 7]. External validity is an assessment of whether the causal estimates deduced from experimental research would persist in other settings and with other samples. For instance, concerns about the external validity of research conducted using student samples (the so-called college sophomore problem) have been debated extensively [18].

The composition of population samples found on crowd-sourcing sites such as MECHANICAL TURK generally differs markedly from the overall population, leading some researchers to question the overall value of online surveys [3, 6, 10, 12, 28, 40, 48, 49, 58, 59, 87, 94, 97]. Our stance is considerably more optimistic. We believe, with appropriate bias correction techniques, valid results can in fact be attained.

Wauthier *et al.* [120] advocate a bias correction approach for crowd-sourced data that we generally follow.

Example 8 (Unbiasing) Consider deciding if there are more female liberal art majors than than the there are male ones. The ultimate comparison will be performed via a t-test. However, the first task is to determine the expected value of female and male liberal art majors given that we drew S samples from the crowd.

These values can be computed as shown below:

$$E[L_W|C] = Pr[L|W_C] \times Pr[W_C|W_W] \times S$$

$$E[L_M|C] = Pr[L|M_C] \times Pr[M_C|M_W] \times S$$

where L_W and L_M are the number of female and male liberal art major, respectively, W_C and M_C stand for a woman/man being in the crowd, and W_W and M_W stand for a woman/man being the world as reflected by a broad population survey such as the US census; the latter two probabilities may be related at 51 : 49, for example.

Note that our goal is to discern the expected value of liberal art majors per gender *in the world*. We can unbias

our data by using the probability of observing a woman in the crowd given there is a woman in the world:

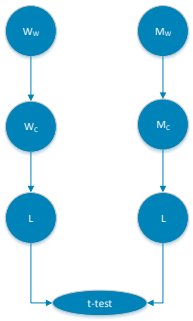
$$E[W_L|W] = E[W_L|C] \times P(W_C|W_W)$$

and similarly for men

$$E[M_L|M] = E[M_L|C] \times P(M_C|M_W).$$

While $E[W_L|C]$ and $E[M_L|C]$ can be approximated by observing the crowd-sourced results for the female and male sub-segments of the population, coefficients such as $P(W_C|W_W)$ can be computed from our knowledge of crowd population vs. that in the world in general. For example, if women to men are at 50%:50% in the world and at 30%:70% in the crowd, $P(W_C|W_W) = .7$ and $P(M_C|M_W) = .3$. □

Note that the above example presents a simple model that does not, for example, explicitly represent the factor of ignorability [38], pg. 202 of our experimental design. Also note that unbiasing generally may need to be done before we perform a t-test to reshape the underlying distributions.



2.4 Power Analysis

Power analysis is necessary to determine the number of samples we need to acquire for the query or queries to have statistical significance. In general, this corresponds to the number of workers we need to obtain. In InterPoll we use the Bayesian approach to power analysis, as outlined in several papers by Kruschke [62–65].

2.5 Back-end Platform Selection

In principle, InterPoll can run on any crowd-sourcing platform, although for specialized tasks we may prefer mobile back-ends or back-ends that support special equipment (such as a crowd of FitBit or wearers who are willing to share their data).

Given the nature of common polling tasks, it is generally cheaper to use general labor markets such as Mechanical Turk or CrowdFlower. It may be further desirable to specialize for a workforce that is accustomed to answering surveys, such as that maintained by Google Customer Surveys or Instant.ly, however, generally workers naturally learn the kind of tasks a particular task creator publishes and gravitate to those, creating a more consistent labor pool. Our chief requirements for the back-end are:

- We can obtain demographic information for workers;
- Back-end interactions can be scripted.

Both turn out to be requirements that are no so easy to fulfill. Survey platforms such as Instant.ly do not allow easy scripting and do not provide SDKs.

Our default implementation in InterPoll, we use Mechanical Turk as our back-end of choice and explicitly ask demographic questions (such as gender, age, income, etc.) of the participants. Unfortunately, at the moment we do not have any way to verify the truthfulness of the answers provided [52, 86]. This is something that could be alleviated with platform support, which could, for instance, cross-correlate provided geographical information with that on the worker's payment credentials.

A separate theme within InterPoll is that of selecting the right back-end crowd for a particular query. Back-end choices are largely influenced by the cost model that is supported by the back-end, as well as the overall number of participating workers, the demand for work at a particular price level, and the latency of task completion [81]. Our chief focus in InterPoll is on the monetary cost of executing queries, although the other secondary criteria are obviously important as well.

3. Related Work

There are several bodies of related work from fields that are usually not considered to be particularly related, as outline below.

3.1 Crowd-Sourcing Systems

There has been a great deal of interest in recent years in building new systems for automating crowd-sourcing tasks.

Toolkits: TurKit [71] is one of the first attempts to automate programming crowd-sourced systems. Much of the focus of TurkIt is the iterative paradigm, where solutions to crowd-sourced tasks are refined and improved by multiple workers sequentially. The developer can write TurkIt scripts using JavaScript. AutoMan [4] is a programmability approach to combining crowd-based and regular programming tasks, a goal shared with Truong *et al.* [114]. The focus of AutoMan is on computation reliability, consistency and accuracy of obtained results, as well as task scheduling. Turkomatic [66, 67] is a system for expression crowd-sourced tasks and designing workflows. CrowdForge is a general purpose framework for accomplishing complex and interdependent tasks using micro-task markets [60]. Some of the tasks involve article writing, decision making, and science journalism, which demonstrates the benefits and limitations of the chosen approach. More recently, oDesk has emerged as a popular marketplace for skilled labor. CrowdWeaver is a system to visually manage complex crowd work [57]. The system supports the creation and reuse of crowd-sourcing and computational tasks into integrated task flows, manages the flow of data between tasks, etc.

Wiki surveys [95] is a novel approach of combining surveys and free-form interviews to come up to answers to tough questions. These answers emerge as a result of pair-wise comparisons of individual ideas volunteered by participants. As an example, participants in the wiki survey were presented with a pair of ideas (e.g., "Open schoolyards across the city as public playgrounds" and "Increase targeted tree plantings in neighborhoods with high asthma rates"), and asked to choose between them, with subsequent data analysis employed to estimate "public opinion" based on a large number of pair-wise outcomes.

We do not aim to adequately survey the vast quantity of crowd-sourcing-related research out there; the interested reader may consult [126]. Notably, a great deal of work has focused on matching users with tasks, quality control, decreasing the task latency, etc.

Moreover, we should note that our focus is on *opinion polls* which distinguishes InterPoll work from the majority of crowd-sourcing research which requires given solution to a particular task such as deciphering a license plate number in a picture, translating sentences, etc. In InterPoll, we are primarily interested in self-reported opinions of users about themselves and their preferences.

Some important verticals: Some crowd-sourcing systems choose to focus on specific verticals. The majority of literature focuses on the following four verticals:

- social sciences [3, 5, 12, 15, 18, 30, 40, 59, 90];
- political science and election polls [5–7, 53, 103, 106, 125];
- marketing [28, 47, 117]; and
- health and well-being [2, 5, 7, 21, 29, 92, 97, 109, 110, 124].

3.2 Optimizing Crowd Queries

CrowdDB [33] uses human input via crowd-sourcing to process queries that regular database systems cannot adequately answer. For example, when information for IBM is missing in the underlying database, crowd workers can quickly look it up and return as part of query results, as requested. CrowdDB uses SQL both as a language for posing complex queries and as a way to model data. While CrowdDB leverages many aspects of traditional database systems, there are also important differences. CrowdDB extends a traditional query engine with a small number of operators that solicit human input by generating and submitting work requests to a microtask crowd-sourcing platform. It allows any column and any table to be marked with the `CROWD` keyword. From an implementation perspective, human-oriented query operators are needed to solicit, integrate and cleanse crowd-sourced data. Supported crowd operators include *probe*, *join*, and *compare*.

Marcus *et al.* [74–76] have published a series of papers outlining a vision for Qurk, a crowd-based query system for managing crowd workflows. Some of the motivating examples [75] include identifying people in photographs, data discovery and cleaning (who is the CEO of a particular company?), sentiment identification in Twitter messages, etc.

Qurk implements a number of optimizations [76], including task batching, replacing pairwise comparisons with numerical ratings, and pre-filtering tables before joining them, which dramatically reduces the overall cost of sorts and joins on the crowd. End-to-end experiments show cost reductions of $14.5x$ on tasks that involve matching up photographs and ordering geometric pictures. These optimization gains in part inspire our focus on cost-oriented optimizations in InterPoll.

Marcus *et al.* [74] study how to estimate the *selectivity* of a predicate with help from the crowd, such as filters photos of people to those of males with red hair. Crowd workers are shown pictures of people and provide either the gender or hair color they see. Suppose we could estimate that red hair is prevalent in only 2% of the photos, and that males constitute 50% of the photos. We could order the tasks to ask about red hair first and perform fewer HITs overall. Whereas traditional selectivity estimation saves database users time, optimizing operator ordering can save users money by reducing the number of HITs. We consider these estimation techniques very much applicable to the setting of InterPoll, especially when it comes to free-form `PoseQuestion`, where we have no priors informing us of the selectivity factor of such a filter. We also envision of a more dynamic way to unfold questions in an order optimized for cost reduction.

Kittur *et al.* [57] present a system called CrowdWeaver, designed for visually creating crowd workflows. CrowdWeaver system supports the creation and reuse of crowd-sourcing and computational tasks into integrated task flows, manages the flow of data between tasks, and allows tracking and notification of task progress. While our focus in InterPoll is on embedding polls into general-purpose programming languages such as C#, InterPoll could definitely benefit from a visual task builder approach, so we consider CrowdWeaver complimentary.

Somewhat further afield, Gordon *et al.* [39] describe a language for probabilistic programming and give an overview of related work. Nilesh *et al.* [22] talk about *probabilistic databases* designed to work with imprecise data such as measured GPS coordinates and the like.

3.3 Database and LINQ Optimizations

While language-integrated queries are wonderful for bringing the power of data access to ordinary developers, LINQ queries frequently do not result in most efficient executions. There has also been interest in both formalizing the semantics of [16] and optimizing LINQ queries.

Grust *et al.* propose a technique for alternative efficient LINQ-to-SQL:1999 compilation [43]. Steno [83] proposes a strategy for removing some of the inefficiency in built-in LINQ compilation and eliminates it by fusing queries and iterators together and directly compiling LINQ queries to .NET code.

Nerella *et al.* [84] relies on programmer-provided annotations to devise better queries plans for language-integrated queries in JQL, Java Query Language. Annotations can provide information about shapes of distribution for continuous data, for example. Schueller *et al.* [99] focus on bringing the idea of *update propagation* to LINQ queries and combining it with reactive programming. Tawalare *et al.* [112] explore another compile-time optimization approach for JQL.

Bleja *et al.* [9] propose a new static optimization method for object-oriented queries dealing with a special class of sub-queries of a given query called "weakly dependent subqueries". The dependency is considered in the context of SBQL non-algebraic query operators like selection and projection. This research follows the stack-based approach to query languages.

3.4 Web-Based Polls and Surveys

Since the time the web has become commonplace for large segments of the population, there has been an explosion of interest in using it as a means for conducting surveys. Below we highlight but several papers in the growing literature on this subject [2, 5, 19, 20, 23, 24, 28, 31, 32, 36, 40–42, 44, 51, 55, 56, 58, 59, 77, 97, 100, 105, 113, 124].

Online Demographics: Recent studies reveal much about the demographics of crowd-sourcing sites such as Amazon's Mechanical Turk [3, 6, 12, 25, 28, 40, 48, 49, 58, 59, 87, 90, 94, 97, 121]. Berinsky *et al.* [6] investigate the characteristics of samples drawn from the Mechanical Turk population and show that respondents recruited in this manner are often *more* representative of the U.S. population than in-person convenience samples — the modal sample in published experimental political science — but *less* representative than subjects in Internet-based panels or national probability samples. They succeeded in replicating three experiments, the first one of which focuses on welfare spendings or assistance to the poor. They compared Mechanical Turk results with those obtained via the General Social Surveys (GSS), a nationally-representative face-to-face interview sample. While subtle differences exist, the overall results were quite similar between the GSS and Mechanical Turk (37% vs 38%). The second experiment involve replicating the so-called *Asian disease* experiment, which involves asking respondents to choose between two policy options. The results were comparable to those obtained in

the original experiment by Tversky and Kahneman [115] on a student sample. The last experiment is described in Kam *et al.* [101] and involves measuring the preference for a risky policy option over a certain policy option. Additionally, Berinsky *et al.* discuss the internal and external validity threats. These three experiments provide a diverse set of studies to reproduce using InterPoll.

Ipeirotis [48, 49] focuses his analysis on the *demographics* of the Mechanical Turk marketplace. Overall, they find that approximately 50% of the workers come from the United States and 40% come from India. Significantly more workers from India participate on Mechanical Turk because the online marketplace is a primary source of income, while in the US most workers consider Mechanical Turk a secondary source of income. While money is a primary motivating reason for workers to participate in the marketplace, workers also cite a variety of other motivating reasons, including entertainment and education. Along with other studies, Ipeirotis provides demographic comparisons for common categories such as gender, age, education level, household income, and marital status for both countries. Ipeirotis [49] digs deeper into worker motivation, cost vs. the number of workers interested, time of completion vs. the reward, etc. We believe that this data can be useful to give more fine-grained cost predictions for InterPoll queries and producing more sophisticated query plans involving tasks priced at various levels, for example. Additionally, while our initial focus is on query cost, we should be able to model completion rates fairly precisely as well.

Paolacci *et al.* [87] compare different recruiting methods (lab, traditional web study, web study with a specialized web site, Mechanical Turk) and discuss the various threats to validity. They also present comparisons of Mechanical Turk samples with those found through subject recruitment at a Midwestern university and through several online discussion boards that host online experiments in psychology, revealing drastic differences in terms of the gender breakdown, average age, and subjective numeracy. The percentage of failed catch trials varied as well, but not drastically; Mechanical Turk workers were quite motivated to complete the surveys, compared to those found though online discussion boards. While data quality does not seem to be adversely affected by the task payoff, researcher reputation might suffer as a result of poor worker perception and careless researchers "black-listed" on sites such as `http://turkopticon.differenceengines.com`.

Ross *et al.* [94] describe how the worker population has changed over time, shifting from a primarily moderate-income, U.S.-based workforce towards an increasingly international group, with a significant population of young, well-educated Indian workers. This change in population points to how workers may treat Turking as a full-time job, which they rely on to make ends meet. The paper contains comparisons across nationality, gender, age, and income, pinpointing a trend towards a growing number of young, male, Indian Turkers. Interesting opportunities exist for cost optimizations in InterPoll if we determine that different worker markets can provide comparable results (for a given query), yet are priced differently.

Buhrmester *et al.* [12] report that demographic characteristics suggest that Mechanical Turk participants are at least as diverse and more representative of non-college populations than those of typical Internet and traditional samples. Most importantly, we found that the quality of data provided by Mechanical Turk met or exceeded the psychometric standards associated with published research.

Andreson *et al.* [1] report that Craigslist can be useful in recruiting women and low-income and young populations, which are often underrepresented in surveys, and in recruiting a racially representative sample. This may be of particular interest in addressing recruitment issues in health research and for recruiting non-WEIRD (Western, Educated, Industrialized, Rich, Democrat) research subjects [46].

Online vs. Offline: Several researchers have studied the advantages and disadvantages of web-based vs. telephone or other traditional survey methodologies [2, 25, 34, 102, 106, 123, 125], with Dillman [23] providing a book-length overview. Sinclair *et al.* [102] focus on epidemiological research, which frequently requires collection of data from a representative sample of the community, or recruitment of specific groups through broad community approaches. They look at response rates for mail and telephone surveys, but web surveys they consider involve direct mailing of postcards and inviting recipients to fill out an online survey and as such do not provide compelling incentives compared to crowd-sourced studies. Fricker [34] compare telephone and Web versions of a questionnaire that assessed attitudes toward science and knowledge of basic scientific facts. However, again, the setting differs significantly from that of InterPoll, in that crowd workers have a direct incentive to participate and complete the surveys.

Duffy [25] give a comparison of online and face-to-face surveys. Issues studies include interviewer effect and social desirability bias in face-to-face methodologies; the mode effects of online and face-to-face survey methodologies, including how response scales are used; and differences in the profile of online panelists, both demographic and attitudinal. Interestingly, Duffy *et al.* report questions pertaining to technology use should not be asked online, as they result in much higher use numbers (i.e., *PC use at home* is 91% in the online sample vs. 53 in the face-to-face sample). Surprisingly, these differences pertain even to technologies such as DVD players and digital TV. They also conclude that online participants are frequently better informed about issues such as cholesterol, and are likely to quickly search for an answer, which compromises the ability to ask knowledge-based questions, especially in a crowd setting. Another conclusion is that for online populations, propensity score weighting has a significant effect, especially for politically-oriented questions.

Stephenson *et al.* [106] study the validity of using online surveys vs. telephone polls by examining the differences and similarities between parallel Internet and telephone surveys conducted in Quebec after the provincial election in 2007. Both samples have demographic characteristics differing slightly, even after re-weighting, from the that of the overall population. Their results indicate that the responses obtained in each mode differ somewhat, but that few inferential differences would occur depending on which dataset were used, highlighting the attractiveness of online surveys, given their generally lower cost.

Biases: Biases in online crowds, compared to online populations, general populations as well as population samples obtained via different recruitment techniques have attracted a lot of attention [3, 53, 89–92, 96], but most conclusions have been positive. In particular, crowds often provide more diversity of participants, on top of higher completion rates and frequently quality of work.

Antin *et al.* [3] study the *social desirability bias* on Mechanical Turk. They use a survey technique called *the list*

experiment which helps to mitigate the effect of social desirability on survey self-reports. Social desirability bias refers to "the tendency of people to deny socially undesirable traits or qualities and to admit to socially desirable ones" [17]. Among US Turkers, they conclude that social desirability encourages over-reporting of each of four motivating factors examined; the over-reporting was particularly large in the case of money as a motivator. In contrast, among Turkers in India we find a more complex pattern of social desirability effects, with workers under-reporting "killing time" and "fun" as motivations, and drastically over-reporting "sense of purpose."

Survey sites: In the last several years, we have seen surveys sites that are crowd-backed. The key distinction between these sites and InterPoll is our focus on optimizations and statistically significant results at the lowest cost. In contrast, survey sites generally are incentivized to encourage the survey-maker to solicit as many participants as possible . At the same time, we draw inspiration from many useful features that the sites described below provide.

Most survey cites give easy access to non-probability samples of the Internet population, generally without attempting to correct for the inherent population bias. Moreover, while Internet use in the United States is approaching 85% of adults, users tend to be younger, more educated, and have higher incomes [88]. Unlike other tools we have found, Google Customer Surveys support re-weighting the survey results to match the deomographics of the Current Population Survey (CPS) [116].

SurveyMonkey claims to be the most popular survey building platform [47]. In recent years, they have added support for data analytics as well as an on-demand crowd. Market research seems to be the niche they are trying to target [108]. SurveyMonkey performs ongoing monitoring of audience quality through comparing the answers they get from their audience to that obtained via daily Gullop telephone polls [107]. They conclude that the SurveyMonkey Audience 3-day adjusted average, for 5 consecutive days is within a 5% error margin of Gallup's 14-day trailing average. In other words, when corrected for a higher average income of SurveyMonkey respondents in comparison to the US census data, SurveyMonkey is able to produce effectively the same results as Gallup, with only 3-days of data instead of 14 for Gallup.

Instant.ly and uSamp [117] focus primarily on marketing studies and boast an on-demand crowd with very fast turn-around times: some survey are completed in minutes. In addition to rich demographic data, uSamp collects information on the industry in which respondents are employed, their mobile phone type, job title, etc., also allowing to

Unlike other sites, Google Surveys results have been studied in academic literature. McDonald *et al.* [79] compares the responses of a probability based Internet panel, a non-probability based Internet panel, and Google Consumer Surveys against several media consumption and health benchmarks, leading the authors to conclude that despite differences in survey methodology, Consumer Surveys can be used in place of more traditional Internet-based panels without sacrificing accuracy.

Keeter *et al.* [54] present a comparison of results performed at Pew to those obtained via Google Customer Surveys. Note that demographic characteristics for survey-takes appear to be taken from DoubleClick cookies and are generally inferred and not verified (an approach taken by Instant.ly). A clear advantage of this approach is asking fewer questions; however, there are obvious disadvantages.

Apparently, for about 40% of survey-takes, reliable demographic information cannot be determined. The Google Consumer Survey method samples internet users by selecting visitors to publisher websites that have agreed to allow Google to administer one or two questions to their users. As of 2012, there are about 80 sites in the *Google Surveys publisher network* (and 33 more currently in testing). The selection of surveys for eligible visitors of these sites appears random. Details on the Google Surveys "survey wall" appear scarce [26].

The Pew study attempted to validate the inferred demographic characteristic and concluded that for 75% of respondents, the inferred gender matched their survey response. For age inference, the results were mixed, with about 44% confirming the automatically inferred age range. Given that the demographic characteristics are used to create a stratified sample, and to re-weight the survey results, these differences may lead to significant errors; for instance, fewer older people using Google Consumer Surveys approved of Obama's job performance than in the Pew Research survey. The approach taken in InterPoll is to *ask* the user to provide their demographic characteristics; we would immensely benefit from additional support on the back-end level to obtain or verify the user-provided data. Google Customer Surveys have been used for information political surveys [61].

The Pew report concludes that, demographically, the Google Consumer Surveys sample appears to conform closely to the demographic composition of the overall internet population. From May to October 2012, the Pew Research Center compared results for 48 questions asked in dual frame telephone surveys to those obtained using Google Consumer Surveys. Questions across a variety of subject areas were tested, including: demographic characteristics, technology use, political attitudes and behavior, domestic and foreign policy and civic engagement. Across these various types of questions, the median difference between results obtained from Pew Research surveys and using Google Consumer Surveys was 3 percentage points. The mean difference was 6 points, which was a result of several sizable differences that ranged from 10–21 points and served to increase the mean difference. It appears, however, that Google Survey takers are no more likely to be technology-savvy than an average Internet user, largely eliminating that bias. A key limitation for large-scale survey appears to be the inability to ask more than a few questions at a time, which is a limitation of their format [26], and the inability to administer questions to the same responder over time. The focus in InterPoll is on supporting as many questions as the developer wants to include.

SocialSci (`http://www.socialsci.com`) is a survey site specializing in social science studies. On top of features present in other platforms, it features dynamic workflows for complex surveys, a vetting system for survey-takers based on credit ratings, many demographic characteristics, deceit pools, IRB assistance, etc. We are not aware of demographic studies of the SocialSci respondent population.

Statistical Techniques for Surveys: The issue of statistical validity in the context of surveys has long been of interest to statisticians and social science researchers. Two main schools of thought are prominent: the so-called *frequentist* view and the newer, albeit gaining popularity Bayesian view [11, 14, 27, 45, 68–70, 72, 73, 90, 98, 118, 119, 122]. In In-

terPoll, we generally model our approach to bias correction and power analysis on Wauthier *et al.* [120].

4. Conclusions

This paper presents a vision for InterPoll, a language integrated approach to programming crowd-sourced polls. While much needs to be done to achieve the goals outlined in Section 1, we envision InterPoll as a powerful system, useful in a range of domains, including social sciences, political and marketing polls, and health surveys.

References

[1] S. Anderson, S. Wandersee, A. Arcenas, and L. Baumgartner. Craigslist samples of convenience: recruiting hard-to-reach populations.

[2] D. Andrews, B. Nonnecke, and J. Preece. Electronic survey methodology: A case study in reaching hard-to-involve Internet users. *International Journal of . . .*, 2003.

[3] J. Antin and A. Shaw. Social desirability bias and self-reports of motivation: a study of Amazon Mechanical Turk in the US and India. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2012.

[4] D. Barowy, C. Curtsinger, E. Berger, and A. McGregor. AutoMan: A platform for integrating human-based and digital computation. *Proceedings of the ACM international conference on Object oriented programming systems languages and applications - OOPSLA '12*, page 639, Jan. 2012.

[5] T. S. Behrend, D. J. Sharek, and A. W. Meade. The viability of crowdsourcing for survey research. *Behavior research methods*, Jan. 2011.

[6] A. Berinsky, G. Huber, and G. Lenz. Evaluating Online Labor Markets for Experimental Research: Amazon.com's Mechanical Turk. *Political Analysis*, 20(3):351–368, July 2012.

[7] A. J. A. Berinsky, G. A. G. Huber, and G. S. Lenz. Using mechanical Turk as a subject recruitment tool for experimental research. *Typescript, Yale*, pages 1–26, 2010.

[8] S. J. Best and B. S. Krueger. *Exit Polls: Surveying the American Electorate, 1972-2010*. 2012.

[9] M. Bleja, T. Kowalski, and K. Subieta. Optimization of object-oriented queries through rewriting compound weakly dependent subqueries. *Database and Expert Systems*, pages 1–8, Jan. 2010.

[10] J. Bornholt, T. Mytkowicz, and K. S. Mckinley. Uncertain<T>: A first-order type for uncertain data. 2013.

[11] F. Bourguignon and M. Fournier. Selection bias corrections based on the multinomial logit model: Monte Carlo comparisons. *of Economic Surveys*, Jan. 2007.

[12] M. Buhrmester and T. Kwang. Amazon's Mechanical Turk: A new source of inexpensive, yet high-quality, data? *on Psychological Science*, Jan. 2011.

[13] T. D. Buskirk, D. Ph, and C. Andrus. Online Surveys Aren ' t Just for Computers Anymore ! Exploring Potential Mode Effects between Smartphone and Computer-Based Online Surveys. pages 5678–5691, 2010.

[14] M. Callegaro and C. DiSogra. Computing response metrics for online panels. *Public Opinion Quarterly*, Jan. 2008.

[15] J. Chandler, P. Mueller, and G. Paolacci. Methodological concerns and advanced uses of crowdsourcing in psychological research.

[16] J. Cheney, S. Lindley, and P. Wadler. A practical theory of language-integrated query. *Proceedings of the 18th ACM SIGPLAN international conference on Functional programming - ICFP '13*, page 403, Jan. 2013.

[17] D. L. Clancy and K. Phillips J. Some effects of "Social desirability" in survey studies. *The American Journal of Sociology*, 77(5):921–940, 1972.

[18] C. Cooper, D. M. McCord, and A. Socha. Evaluating the college sophomore problem: the case of personality and politics. *The Journal of psychology*, 145(1):23–37, 2011.

[19] M. Couper. Designing effective web surveys, 2008.

[20] M. P. Couper. Review: Web surveys: A review of issues and approaches. *The Public Opinion Quarterly*, pages 1–31, Jan. 2000.

[21] F. Curmi and M. A. Ferrario. Online sharing of live biometric data for crowd-support: Ethical issues from system design. 2013.

[22] N. Dalvi, C. Ré, and D. Suciu. Probabilistic Databases: Diamonds in the Dirt. *Communications of the ACM*, 2009.

[23] D. Dillman, R. Tortora, and D. Bowker. Principles for constructing Web surveys. 1998.

[24] M. Duda and J. Nobile. The fallacy of online surveys: No data are better than bad data. *Human Dimensions of Wildlife*, 2010.

[25] B. Duffy, K. Smith, and G. Terhanian. Comparing data from online and face-to-face surveys. *International Journal of*, Jan. 2005.

[26] J. Ellis. How Google is quietly experimenting in new ways for readers to access publishers' content, 2011.

[27] D. Erceg-Hurn and V. Mirosevich. Modern robust statistical methods: an easy way to maximize the accuracy and power of your research. *American Psychologist*, 2008.

[28] J. Evans, N. Hempstead, and A. Mathur. The value of online surveys. *Internet Research*, 15(2):195–219, Jan. 2005.

[29] J. Eysenbach, G. Eysenbach, and J. Wyatt. Using the Internet for Surveys and Health Research. *Journal of Medical Internet Research*, 4(2):e13, Jan. 2002.

[30] E. Ferneyhough. Crowdsourcing Anxiety and Attention Research, 2012.

[31] K. Fort, G. Adda, and K. B. Cohen. Amazon Mechanical Turk: Gold mine or coal mine? *Computational Linguistics*, pages 1–8, Jan. 2011.

[32] F. Fowler. Survey research methods. 2009.

[33] M. Franklin, D. Kossmann, T. Kraska, S. Ramesh, and R. Xin. CrowdDB: answering queries with crowdsourcing. *SIGMOD '11: Proceedings of the 2011 ACM SIGMOD International Conference on Management of data*, pages 1–12, June 2011.

[34] S. Fricker, M. Galesic, R. Tourangeau, and T. Yan. An experimental comparison of web and telephone surveys. *Public Opinion Quarterly*, 2005.

[35] G. Fritchey. *SQL Server Execution Plans*. Simple Talk Publishing, 2009.

[36] M. Fuchs. Mobile Web Survey: A preliminary discussion of methodological implications. *Envisioning the survey interview of the future*, Jan. 2008.

[37] M. Fuchs and B. Busse. The Coverage Bias of Mobile Web Surveys Across European Countries. 4(1):21–33, 2009.

[38] A. Gelman, J. B. Carlin, H. S. Stern, D. B. Dunson, A. Vehtari, and D. B. Rubin. *Bayesian Data Analysis*. CRC Press, 3rd edition, 2014.

[39] A. D. Gordon, J. Borgstr, N. Rolland, and J. Guiver. Tabular: A Schema-Driven Probabilistic Programming Language. Technical report, Microsoft Research, 2013.

[40] S. Gosling, S. Vazire, S. Srivastava, and O. John. Should we trust web-based studies? A comparative analysis of six preconceptions about Internet questionnaires. *American Psychologist*, 59(2):93–104, Jan. 2004.

[41] R. Groves. Survey errors and survey costs. 2004.

[42] R. M. Groves, F. J. F. Jr., M. P. Couper, J. M. Lepkowski, E. Singer, and R. T. (Author). *Survey Methodology*. Wiley, 2009.

[43] T. Grust, J. Rittinger, and T. Schreiber. Avalanche-safe LINQ compilation. *Proceedings of the VLDB Endowment*, 3(1-2):162–172, Sept. 2010.

[44] H. Gunn. Web-based surveys: Changing the survey process. *First Monday*, 2002.

[45] J. A. Hanley, A. Negassa, and J. E. Forrester. Statistical analysis of correlated data using generalized estimating equations: an orientation. *American journal of*, Jan. 2003.

[46] J. Henrich, S. J. Heine, and A. Norenzayan. The weirdest

people in the world? *The Behavioral and brain sciences*, 33(2-3):61–83; discussion 83–135, June 2010.
[47] HubSpot and SurveyMonkey. Using online surveys in your marketing. pages 1–43.
[48] P. Ipeirotis. Demographics of Mechanical Turk. *2010*, Jan. 2010.
[49] P. G. Ipeirotis. Analyzing the Amazon Mechanical Turk marketplace. *XRDS: Crossroads*, Jan. 2010.
[50] J. H. Janssen and G. Fitzpatrick. Understanding Heart Rate Sharing: Towards Unpacking Physiosocial Space. pages 859–868, 2012.
[51] R. Jurca and B. Faltings. Incentives for expressing opinions in online polls. *Proceedings of the ACM Conference on Electronic Commerce*, 2008.
[52] A. Kapelner and D. Chandler. Preventing Satisficing in Online Surveys : A "Kapcha" to Ensure Higher Quality Data. 2010.
[53] S. Keeter. The impact of cell phone noncoverage bias on polling in the 2004 presidential election. *Public Opinion Quarterly*, 2006.
[54] S. Keeter, L. Christian, and S. Researcher. A Comparison of Results from Surveys by the Pew Research Center and Google Consumer Surveys. 2012.
[55] P. Kellner. Can online polls produce accurate findings? *International Journal of Market Research*, 2004.
[56] A. Kittur, E. H. Chi, and B. Suh. Crowdsourcing user studies with Mechanical Turk. *Proceedings of the SIGCHI conference on*, Jan. 2008.
[57] A. Kittur, S. Khamkar, P. André, and R. Kraut. CrowdWeaver: Visually Managing Complex Crowd Work. *Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work - CSCW '12*, page 1033, Jan. 2012.
[58] R. Kosara and C. Ziemkiewicz. Do Mechanical Turks dream of square pie charts? *Proceedings BEyond time and errors: novel evaLuation methods for Information Visualization*, 2010.
[59] R. Kraut, J. Olson, M. Banaji, A. Bruckman, J. Cohen, and M. Couper. Psychological Research Online: Report of Board of Scientific Affairs' Advisory Group on the Conduct of Research on the Internet. *American Psychologist*, 59(2):105–117, Jan. 2004.
[60] R. E. Kraut. CrowdForge : Crowdsourcing Complex Work. *UIST*, pages 43–52, 2011.
[61] P. Krugman. What People (DonâĂŹt) Know About The Deficit, Apr. 2013.
[62] J. K. Kruschke. *Doing Bayesian Data Analysis: A Tutorial with R and BUGS*. Academic Press, 2010.
[63] J. K. Kruschke. What to believe: Bayesian methods for data analysis. *Trends in cognitive sciences*, 14(7):293–300, July 2010.
[64] J. K. Kruschke. Introduction to Special Section on Bayesian Data Analysis. *Perspectives on Psychological Science*, 6(3):272–273, May 2011.
[65] J. K. Kruschke. Bayesian estimation supersedes the t-test. 2012.
[66] A. Kulkarni, M. Can, and B. Hartmann. Collaboratively crowdsourcing workflows with turkomatic. *of the ACM 2012 conference on*, Jan. 2012.
[67] A. P. Kulkarni, M. Can, and B. Hartmann. Turkomatic: automatic recursive task and workflow design for mechanical turk. *CHI'11 Extended Abstracts on Human*, Jan. 2011.
[68] E. S. Lee and R. N. Forthofer. *Analyzing complex survey data*. Jan. 2006.
[69] S. Lee. Propensity score adjustment as a weighting scheme for volunteer panel web surveys. *Journal of official statistics*, Jan. 2006.
[70] S. Lee and R. Valliant. Estimation for volunteer panel web surveys using propensity score adjustment and calibration adjustment. *Sociological Methods & Research*, Jan. 2009.
[71] G. Little, L. B. Chilton, M. Goldman, and R. C. Miller. TurKit: tools for iterative tasks on Mechanical Turk. *Proceedings of UIST*, pages 1–2, Jan. 2009.
[72] G. Loosveldt and N. Sonck. An evaluation of the weighting procedures for an online access panel survey. *Survey Research Methods*, Jan. 2008.
[73] T. Lumley. Analysis of complex survey samples. *Journal of Statistical Software*, Jan. 2004.
[74] A. Marcus, D. Karger, S. Madden, R. Miller, and S. Oh. Counting with the crowd. *Proceedings of the VLDB Endowment ,*, 6(2), Dec. 2012.
[75] A. Marcus, E. Wu, Karger, S. R. Madden, and R. C. Miller. Crowdsourced databases: Query processing with people. *2011*, Jan. 2011.
[76] A. Marcus, E. Wu, D. Karger, S. Madden, and R. Miller. Human-powered sorts and joins. *Proceedings of the VLDB Endowment ,*, 5(1), Sept. 2011.
[77] W. Mason and S. Suri. Conducting behavioral research on Amazon's Mechanical Turk. *Behavior research methods*, Jan. 2012.
[78] J. Mayo. *LINQ Programming*. McGraw-Hill Osborne Media, 1 edition, 2008.
[79] P. Mcdonald, M. Mohebbi, and B. Slatkin. Comparing Google Consumer Surveys to Existing Probability and Non-Probability Based Internet Surveys.
[80] B. F. Mcsherry. Privacy Integrated Queries : An Extensible Platform for Privacy-Preserving Data Analysis. *Communications of the ACM*, 53(8):89–97, 2009.
[81] P. Minder, S. Seuken, A. Bernstein, and M. Zollinger. CrowdManager - Combinatorial Allocation and Pricing of Crowdsourcing Tasks with Time Constraints. *Workshop on Social Computing and User Generated Content in conjunction with ACM Conference on Electronic Commerce (ACM-EC 2012)*, 2012.
[82] L. Mo, R. Cheng, Kao, X. Yang, C. Ren, S. Lei, D. Cheung, and E. Lo. Optimizing plurality for human intelligence tasks. *Proceedings of the International Conference on Information and Knowledge Management*, Oct. 2013.
[83] D. Murray, M. Isard, and Y. Yu. Steno: automatic optimization of declarative queries. *Proceedings of the Conference on Programming Language Design and Implementation*, pages 1–11, June 2011.
[84] V. Nerella, S. Madria, and T. Weigert. An Approach for Optimization of Object Queries on Collections Using Annotations. *2013 17th European Conference on Software Maintenance and Reengineering*, pages 273–282, Mar. 2013.
[85] B. Nevarez. *Inside the SQL Server Query Optimizer*. Red Gate Books, 2011.
[86] D. M. Oppenheimer, T. Meyvis, and N. Davidenko. Instructional manipulation checks: Detecting satisficing to increase statistical power. *Journal of Experimental Social Psychology*, 45(4):867–872, July 2009.
[87] G. Paolacci, J. Chandler, and P. Ipeirotis. Running experiments on Amazon Mechanical Turk. *Judgment and Decision*, Jan. 2010.
[88] Pew Research Center. Demographics of Internet users, 2013.
[89] S. J. Phillips, M. Dudík, J. Elith, C. H. Graham, A. Lehmann, J. Leathwick, and S. Ferrier. Sample selection bias and presence-only distribution models: implications for background and pseudo-absence data. *Ecological applications : a publication of the Ecological Society of America*, 19(1):181–97, Jan. 2009.
[90] P. Podsakoff, S. MacKenzie, and J. Lee. Common method biases in behavioral research: a critical review of the literature and recommended remedies. *Journal of Applied Psychology*, 88(5):879–903, 2003.
[91] Ramo and S. M. Hall. Reaching young adult smokers through the Internet: Comparison of three recruitment mechanisms. *Nicotine & Tobacco*, Jan. 2010.
[92] D. Ramo, S. Hall, and J. Prochaska. Reliability and validity of self-reported smoking in an anonymous online survey with young adults. *Health Psychology*, 2011.
[93] A. Roshwalb, N. El-Dash, and C. Young. Toward the use of Bayesian credibility intervals in online survey results. 2012.
[94] J. Ross, A. Zaldivar, L. Irani, B. Tomlinson, and M. Silber-

man. Who are the crowdworkers?: shifting demographics in Mechanical Turk. *CHI'10 Extended*, Jan. 2009.
[95] M. Salganik and K. Levy. Wiki surveys: Open and quantifiable social data collection. pages 1–29, Feb. 2012.
[96] L. Sax, S. Gilmartin, and A. Bryant. Assessing response rates and nonresponse bias in web and paper surveys. *Research in higher education*, 2003.
[97] L. Schmidt. Crowdsourcing for human subjects research. *Proceedings of CrowdConf*, 2010.
[98] M. Schonlau, A. Soest, A. Kapteyn, and M. Couper. Selection bias in Web surveys and the use of propensity scores. *Sociological Methods & Research*, 37(3):291–318, Feb. 2009.
[99] G. Schueller and A. Behrend. Stream Fusion using Reactive Programming, LINQ and Magic Updates. *Proceedings of the International Conference on Information Fusion*, pages 1–8, Jan. 2013.
[100] S. Sills and C. Song. Innovations in survey research an application of web-based surveys. *Social science computer review*, 2002.
[101] C. D. Simasa2 and E. N. Kama. Risk Orientations and Policy Frames. *The Journal of Politics*, 72(2), 2010.
[102] M. Sinclair, J. O'Toole, M. Malawaraarachchi, and K. Leder. Comparison of response rates and cost-effectiveness for a community-based survey: postal, internet and telephone modes with generic or personalised recruitment approaches. *BMC medical research methodology*, 12(1):132, Jan. 2012.
[103] N. Sparrow. Developing Reliable Online Polls. *International Journal of Market Research*, 48(6), 2006.
[104] R. Sprou. Exit Polls: Better or Worse Since the 2000 Election? 2008.
[105] J. Sprouse. A validation of Amazon Mechanical Turk for the collection of acceptability judgments in linguistic theory. *Behavior research methods*, Jan. 2011.
[106] L. B. Stephenson and J. Crête. Studying political behavior: A comparison of Internet and telephone surveys. *International Journal of Public Opinion Research*, Jan. 2011.
[107] SurveyMonkey. Data Quality: Measuring the Quality of Online Data Sources. 2012.
[108] SurveyMonkey. Market Research Survey; Get to know your customer, grow your business, 2013.
[109] M. Swan. Crowdsourced health research studies: an important emerging complement to clinical trials in the public health research ecosystem. *Journal of Medical Internet Research*, Jan. 2012.
[110] M. Swan. Scaling crowdsourced health studies : the emergence of a new form of contract research organization. 9:223–234, 2012.
[111] L. Sweeney. k-anonymity: A model for protecting privacy. *International Journal on Uncertainty, Fuzziness and Knowledge-based Systems*, 10(5):1–14, 2002.
[112] S. Tawalare and S. Dhande. Query Optimization to Improve Performance of the Code Execution. *Computer Engineering and Intelligent Systems*, 3(1):44–52, Jan. 2012.
[113] R. Tourangeau, F. G. Conrad, and M. P. Couper. *The Science of Web Surveys*. Oxford University Press, 2013.
[114] H. L. Truong, S. Dustdar, and K. Bhattacharya. Programming hybrid services in the cloud. *Service-Oriented Computing*, pages 1–15, Jan. 2012.
[115] A. Tversky and D. Kahneman. The Framing of Decisions and the Psychology of Choice The Framing of Decisions and the Psychology of Choice. *Science*, 211(4481):453–458, 1981.
[116] US Census. Current population survey, October 2010, school enrollment and Internet use supplement file. (October), 2010.
[117] USamp. Panel Book 2013. 2013.
[118] R. Valliant and J. A. Dever. Estimating propensity adjustments for volunteer Web surveys. *Sociological Methods & Research*, Jan. 2011.
[119] F. Vella. Estimating models with sample selection bias: a survey. *Journal of Human Resources*, 1998.
[120] F. L. Wauthier and M. I. Jordan. Bayesian Bias Mitigation for Crowdsourcing. pages 1–9.
[121] R. W. White. Beliefs and Biases in Web Search. *2013*, Jan. 2013.
[122] C. Winship and L. Radbill. Sampling weights and regression analysis. *Sociological Methods & Research*, Jan. 1994.
[123] K. Wright. Researching Internet-Based Populations: Advantages and Disadvantages of Online Survey Research, Online Questionnaire Authoring Software Packages, and Web Survey Services. *Journal of Computer-Mediated Communication*, 2005.
[124] J. Wyatt. When to use web-based surveys. *Journal of the American Medical Informatics Association*, 2000.
[125] D. Yeager, J. Krosnick, L. Chang, and H. Javitz. Comparing the accuracy of RDD telephone surveys and internet surveys conducted with probability and non-probability samples. *Public Opinion Quarterly*, 2011.
[126] X. Yin, W. Liu, Y. Wang, C. Yang, and L. Lu. What? How? Where? A Survey of Crowdsourcing. *Frontier and Future Development of*, Jan. 2014.
[127] C. Young, J. Vidmar, J. Clark, and N. El-Dash. Our brave new world: blended online samples and performance of no probability approaches.